SOAP
Simple object Access Protocol

マイクロソフト株式会社
シニアテクニカルエバンジェリスト
萩原正義
テクニカルエバンジェリスト
野村一行

SOAP概要

SOAPとは何か?

- 構造化された情報を交換するためのXMLベースの軽量プロトコル
- 単純さと拡張性に重点をおいている
  - プログラミングモデルやアプリケーションのセマンティクスは定義しない
  - 高い拡張性を持つ
- 二つの主なパート
  - データをエンカプセレートするためのエンベロープ
  - アプリケーションで定義されたデータタイプとグラフのインスタンスを表現するためのエンコーディングルール

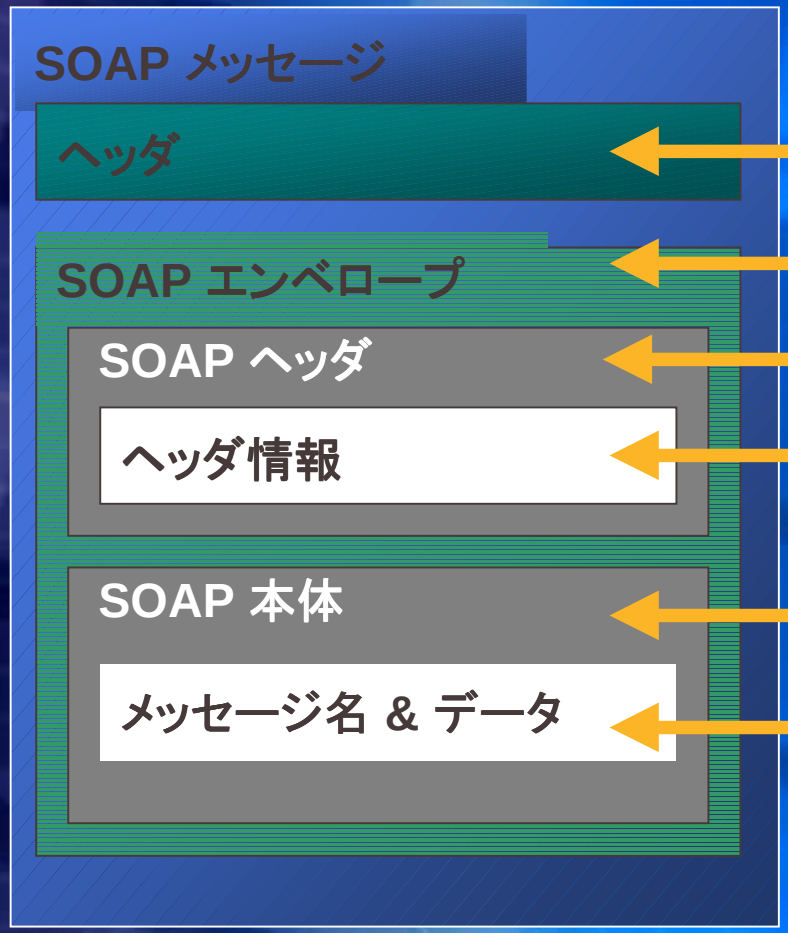
SOAP “onion”

- HTTP マッピングされた SOAP
- メソッド要求インスタンス
- SOAP エンベロープ
- Element Normal Form
- XML スキーマ定義言語 (オプション)
- XML 1.0 + 名前空間

4つの構成要素

- 拡張可能なエンベロープ表現（*必須*）
  - どんな特徴やサービスをメッセージとして表現するか
  - 誰がそれらを扱うか
  - それらは必須か、オプションか
- データのエンコーディングルール（オプション）
  - アプリケーション定義のデータタイプと有向グラフのインスタンス化と交換
  - 実行時に確定される型なども含めたシリアル化の統一モデル
- RPC を表現するための規約（オプション）
  - 呼び出しと応答の方法
- HTTP バインディング（オプション）

SOAPのメッセージ構造



プロトコルバインディングヘッダ

<Envelope> によるペイロード封入

<Header> によるヘッダ宣言

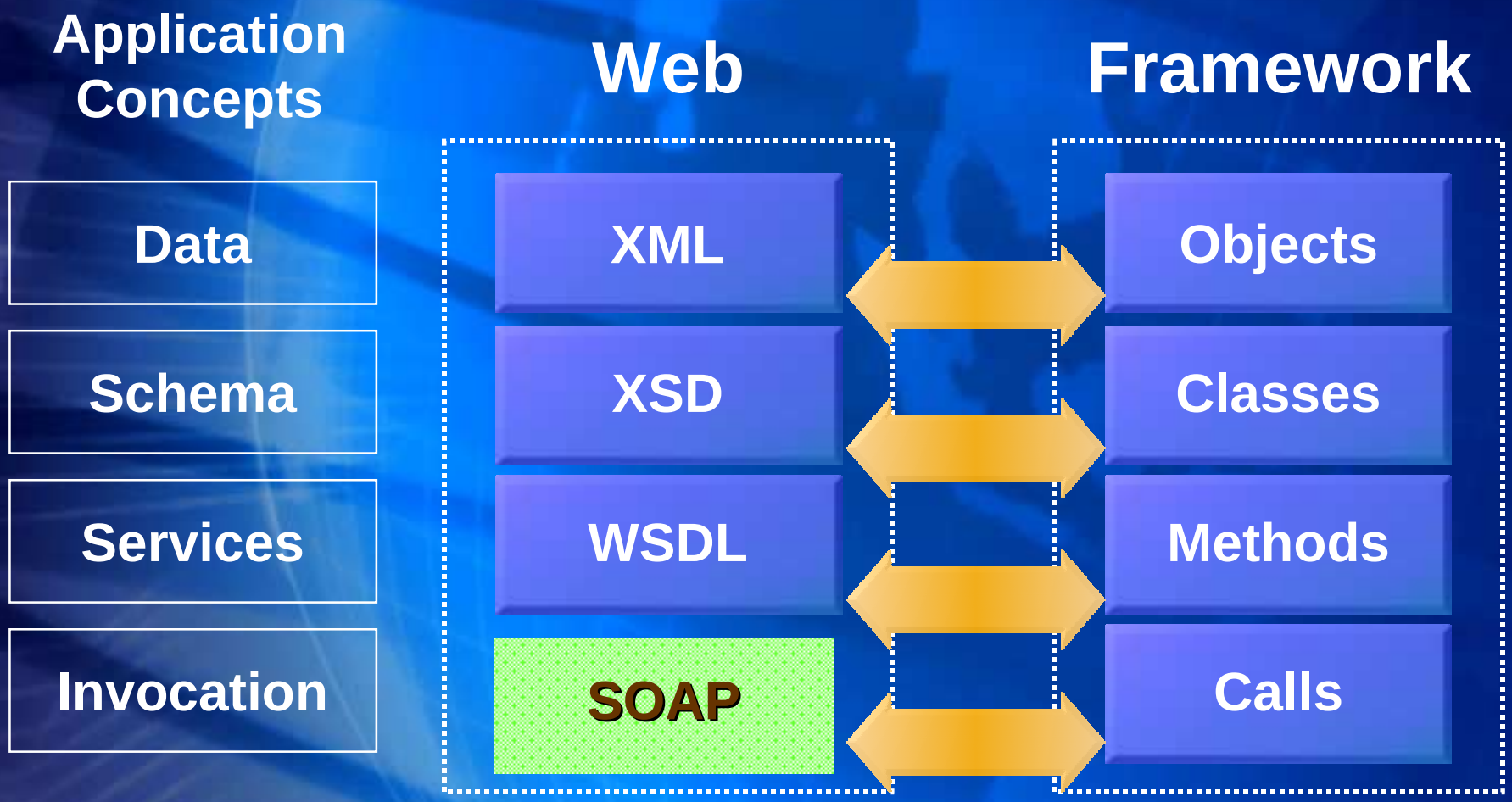
個々のヘッダ

<Body> によるSOAPメッセージ本体の宣言

XMLエンコードされたメッセージ名とデータ

仕様:http://www.w3.org/TR/SOAP/
（日本語訳:http://www.microsoft.com/japan/developer/workshop/xml/general/soapspec.asp）

Web Servicesでの位置付け



SOAPの意義

- ミドルウェアなどに依存しないデータ交換手法の確立：システム間のゆるやかな連携による疎結合化
- 機能が必要な時に機能を探し出して結合する事が実現：L……ate Binding

SOAP への誤解

- SOAP は RPC のみ表現可能
  - SOAP はプログラミングモデルを定義するものではない
  - メッセージング、RPC、分散オブジェクトシステムなどで利用することも可能
- SOAP は HTTP でのみ転送可能
  - SOAP は SOAP エンベロープを乗せるあらゆるプロトコルで転送可能
- SOAP は要求/応答メッセージのみ可能
  - SOAP はメッセージ交換のパターンを定義するものではない
  - SOAP「で」定義、あるいはプロトコルバインドから継承

SOAP HTTP binding

- RPCのHTTP Binding
  - HTTP POSTリクエスト：SOAPリクエスト
  - HTTPレスポンス：SOAPレスポンス
- HTTPヘッダーフィールドの拡張
  - SOAP Action：対象オブジェクトのURIやその他の拡張（WSDLの指示など）
- SOAPヘッダーフィールドの利用
  - SOAPヘッダーでのトランザクションID管理

HTTP Binding

```
POST /path/foo.pl HTTP/1.1
Content-Type: text/xml
SOAPAction: ”interfaceURI#Add ”
Content-Length: nnnn

<SOAP-ENV:Envelope xmlns:SOAP-ENV=”uri for soap”>
  <SOAP-ENV:Body>
    <Add xmlns=”interfaceURI”>
      <arg1>2</arg1>
      <arg2>5</arg2>
    </Add>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```
200 OK
Content-Type: text/xml
Content-Length: nnnn

<SOAP-ENV:Envelope
   xmlns:SOAP-ENV= ”uri for soap” >
  <SOAP-ENV:Body>
    <AddResponse xmlns= ”interfaceURI” >
      <sum>7</sum>
    </AddResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP HTTPの利点

- ファイヤーウォールの通過及びファイヤーウォールによる通信の管理
- スケーラブルなプロキシの配布
- 異なるミドルウェア間でのオブジェクト呼び出し

SOAPセキュリティー

- SSL3によるトランスポートレベルの認証、署名、暗号化
- XMLドキュメントレベルの認証、署名、暗号化
  - 認証：PKIベースの認証
  - エレメントの暗号化：W3C EncryptXMLのサポート
  - エレメントへの署名：W3C XML Dsigのサポート

SOAPの問題

- 相互接続性に関する問題
  - 仕様におけるProtocol Binding（HTTP)部分の不明確さ
    - HTTPプロトコルシーケンス
    - SOAPActionの符号化
  - Actorによるルーティング
  - XMLインスタンスデータのデータ型指定
  - 返り値の要素名の規定がない
  - 少ない相互接続テスト
  - SOAPだけでは完結しないオブジェクト呼び出しの相互接続

SOAPの問題

- 未実装のプロトコル
  - ディレクトリサービス
  - サービスディスクリプション
  - リライアブル
  - セキュリティー
  - トランザクション
  - アタッチメントドキュメントサポート

SOAPが適応する局面と不適合な局面

- 疎結合：Late binding
  - 呼び出し相手が不定
  - データモデルの変更
  - Long transaction（補償トランザクション）
- 密結合
  - トランザクション性能
  - EDIのようなオフセットベースデータモデル
  - 状態の同期（ステートフル）

今後のSOAP

- 各ミドルウェアへの実装と相互接続試験による仕様解釈のすりあわせ
- SOAPをサポートするプロトコルの仕様化
- SOAPの進化：XML Protocol

More Information

- SOAP/1.1 spec
- SOAP W3C Submission Request
- Mailing lists:
  - xml-dist-app@w3.org
    soap@discuss.develop.com

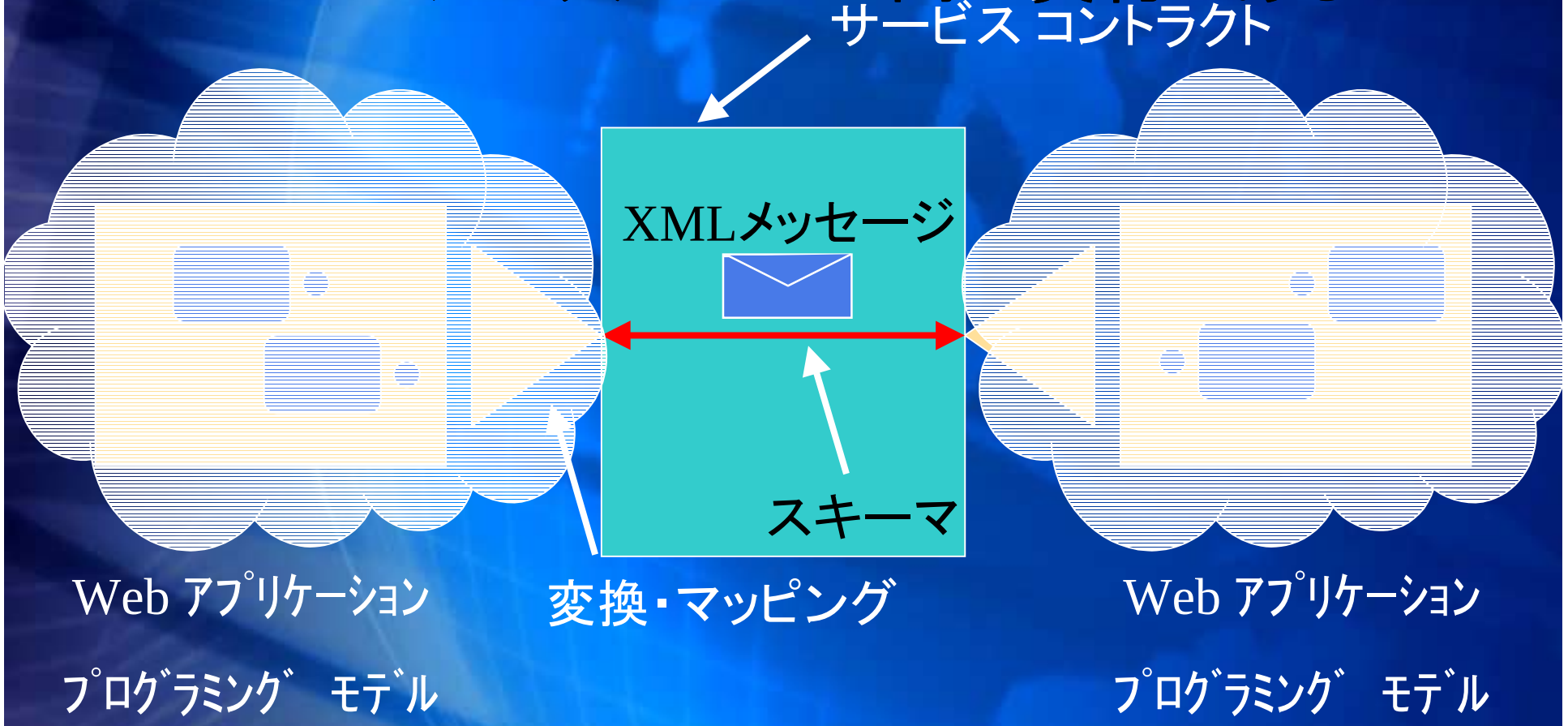
SOAP を利用した Web サービスの構築（SOAP関連技術）

Web アプリケーションの特徴

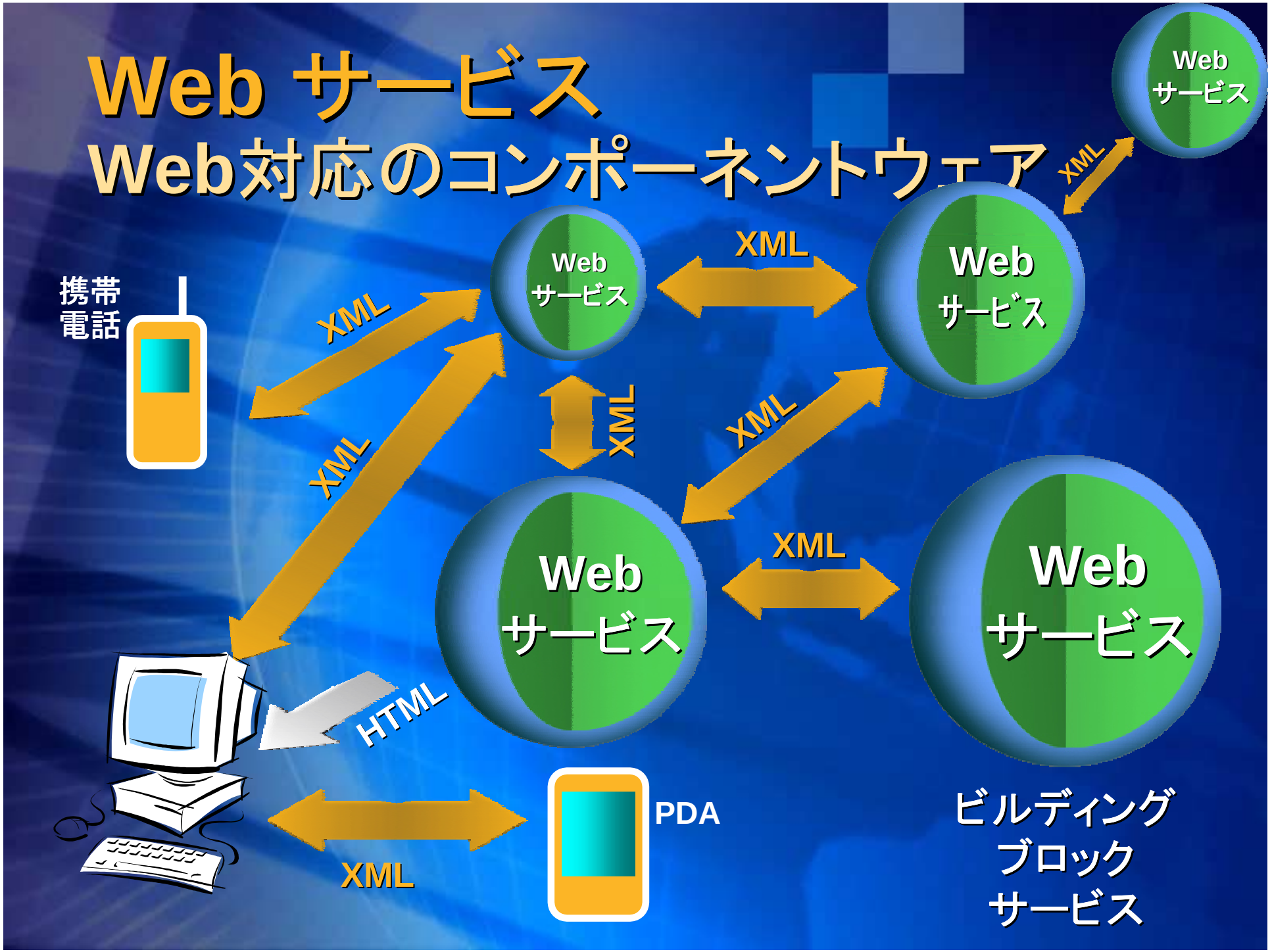
- Web の目的
  - 情報の交換：コラボレーション
  - 問題の解決：サービス
- Web アプリケーションの要件
  - ファイアーウォール越えのメッセージ交換
  - スケーラブル
  - 実行時に他のアプリケーションと統合が可能
  - アプリケーションサービスの検索、発見
  - Time to market の短縮（= 開発期間の短縮）
  - …

サービスのゆるやかな連携

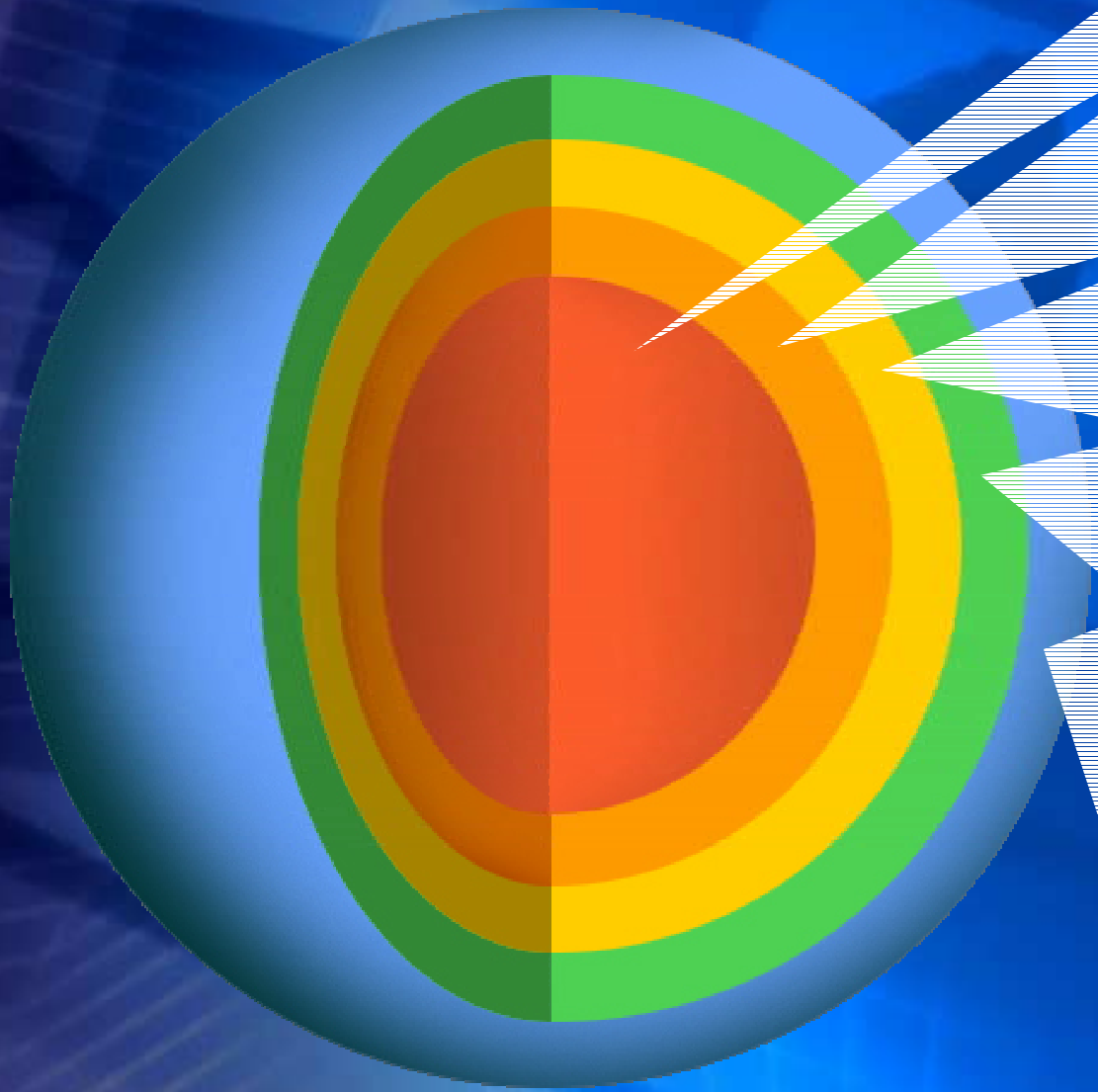
- XMLはアプリケーション間の契約である



Web サービス
Web対応のコンポーネントウェア



Web サービスと標準



コアの XML 標準

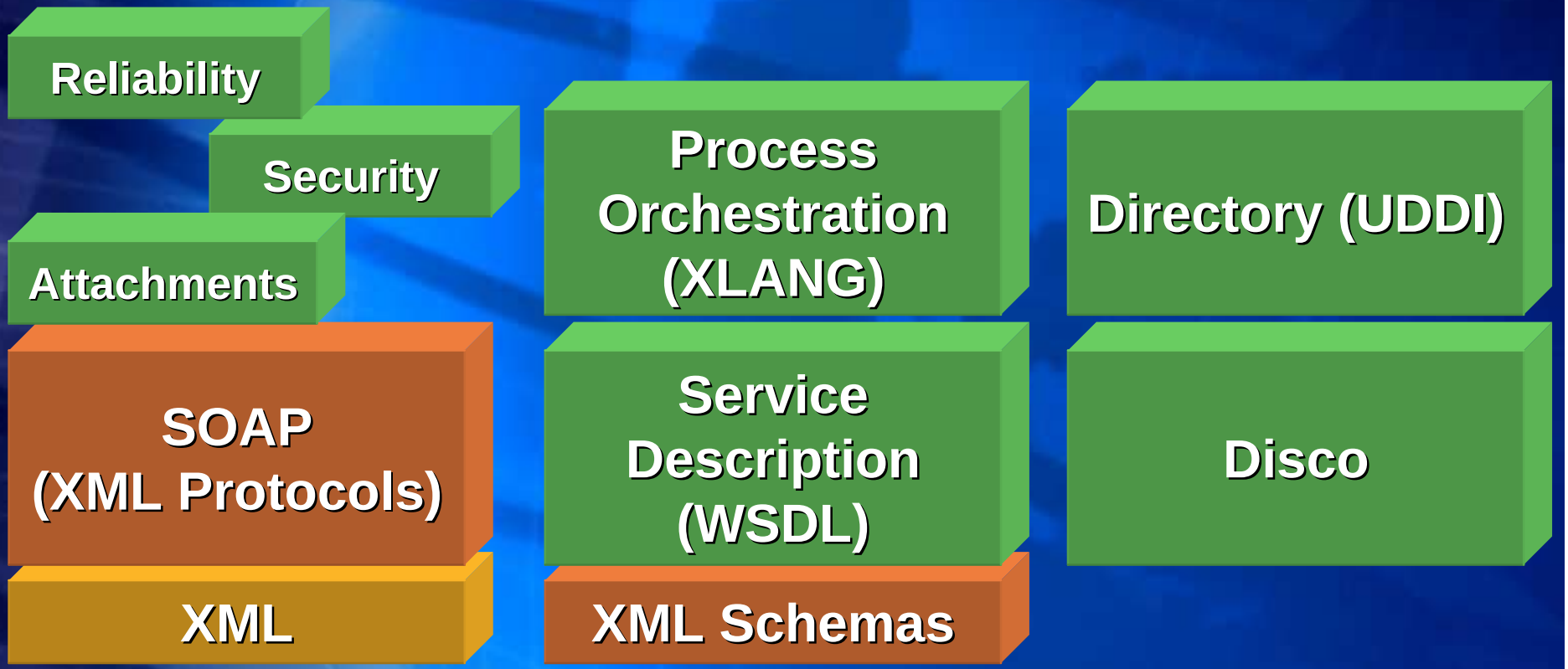
通信インフラ
SOAP、WSDL、UDDI...

アプリケーション スキーマ
アプリ/ドメイン特定

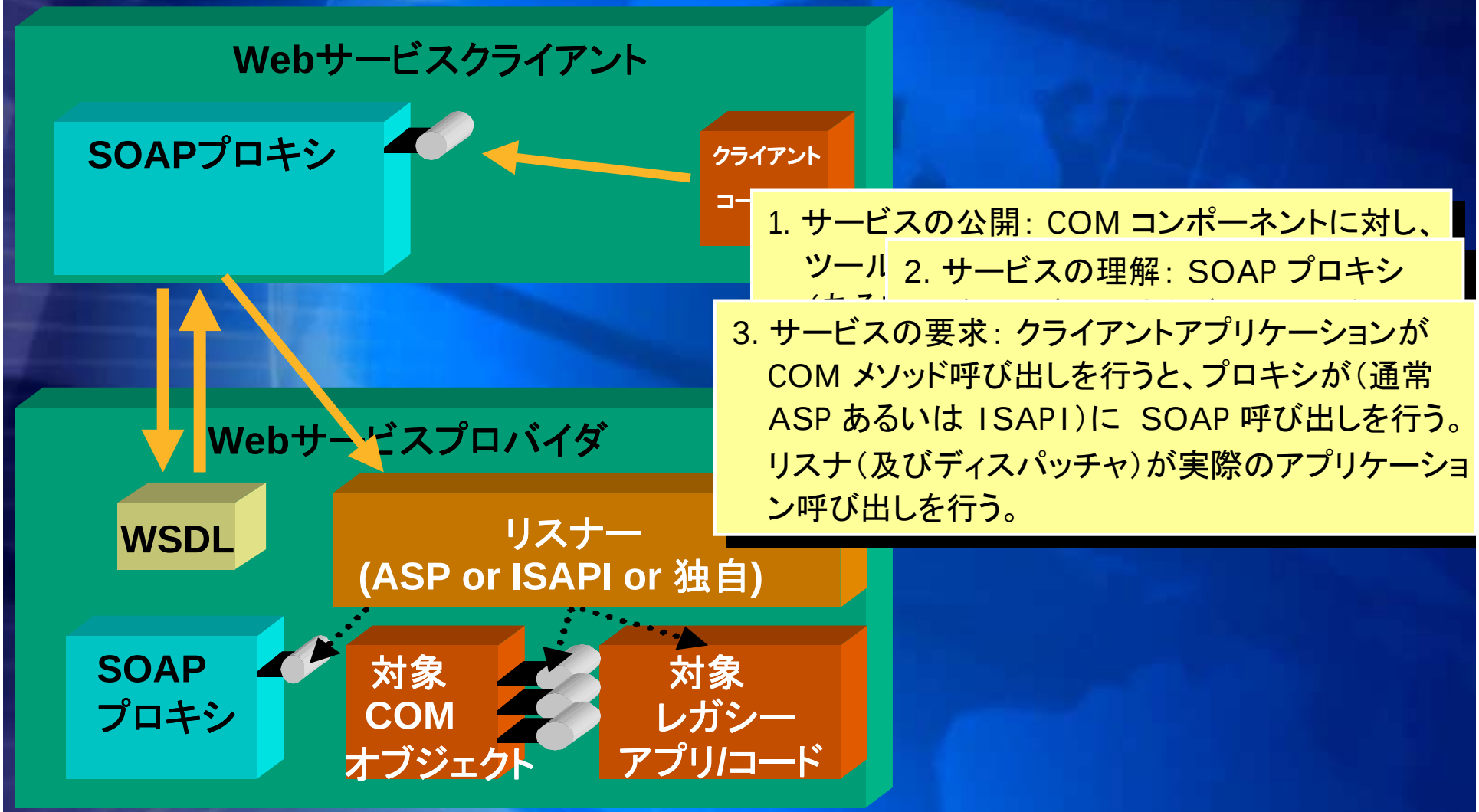
実装
プラットフォーム/ツール

ソリューション、アプリケーション

コミュニケーションのビルディングブロック

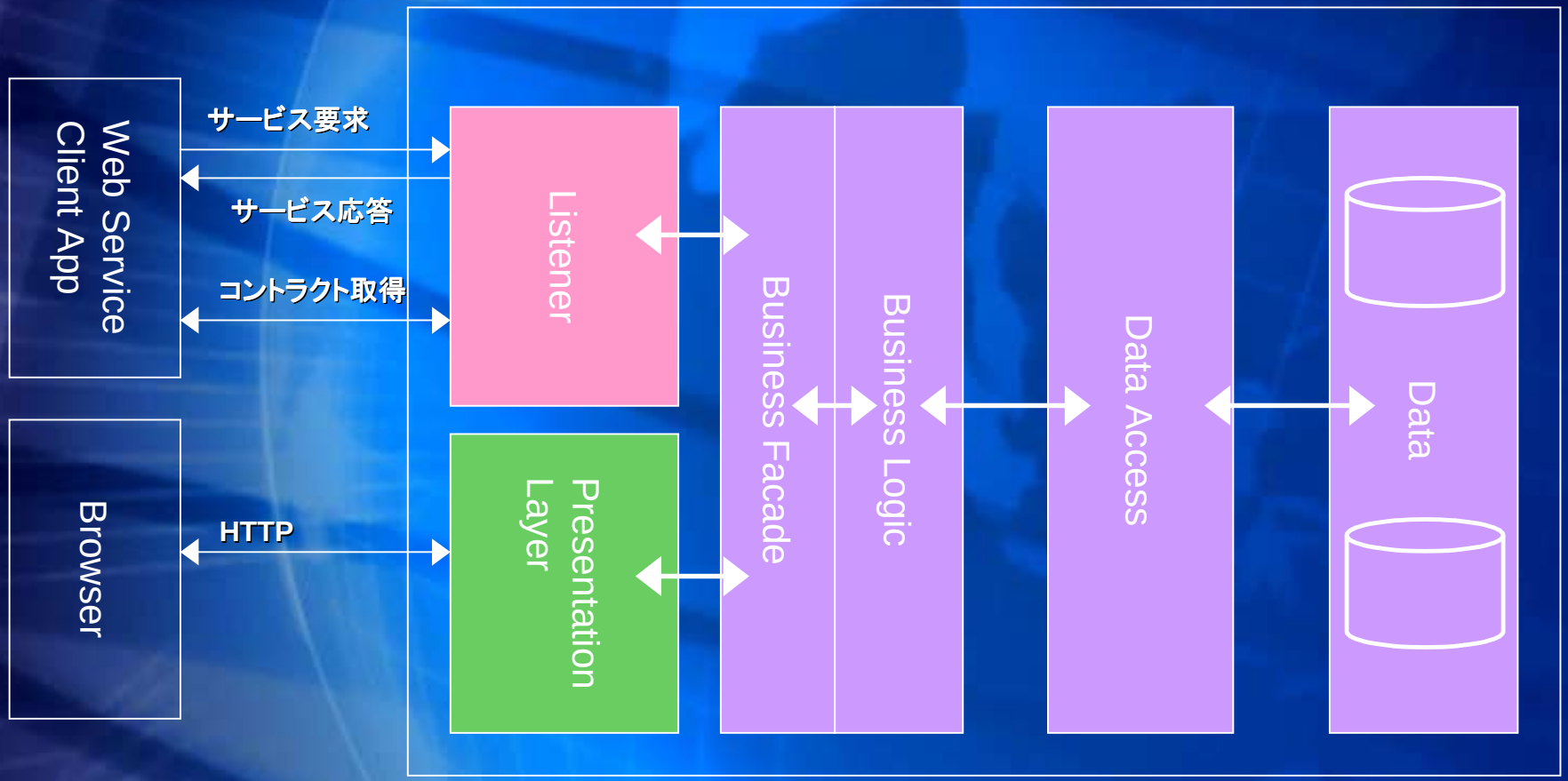


Web サービスの動的な配布

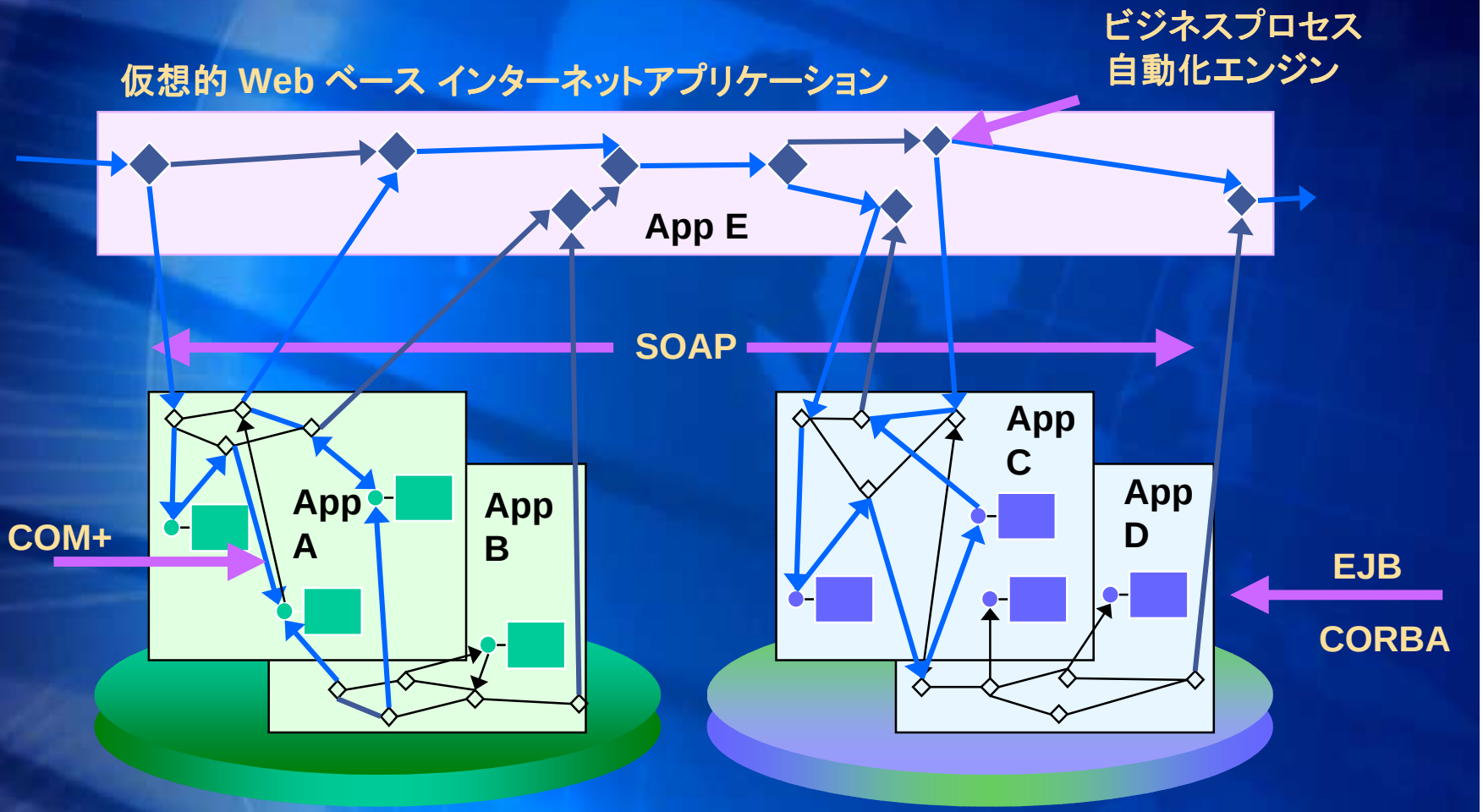


1. サービスの公開：COM コンポーネントに対し、
ツール
2. サービスの理解：SOAP プロキシ
3. サービスの要求：クライアントアプリケーションが COM メソッド呼び出しを行うと、プロキシが（通常 ASP あるいは ISAPI）に SOAP 呼び出しを行う。リスナ（及びディスパッチャ）が実際のアプリケーション呼び出しを行う。

Web サービスの一般的なアーキテクチャ

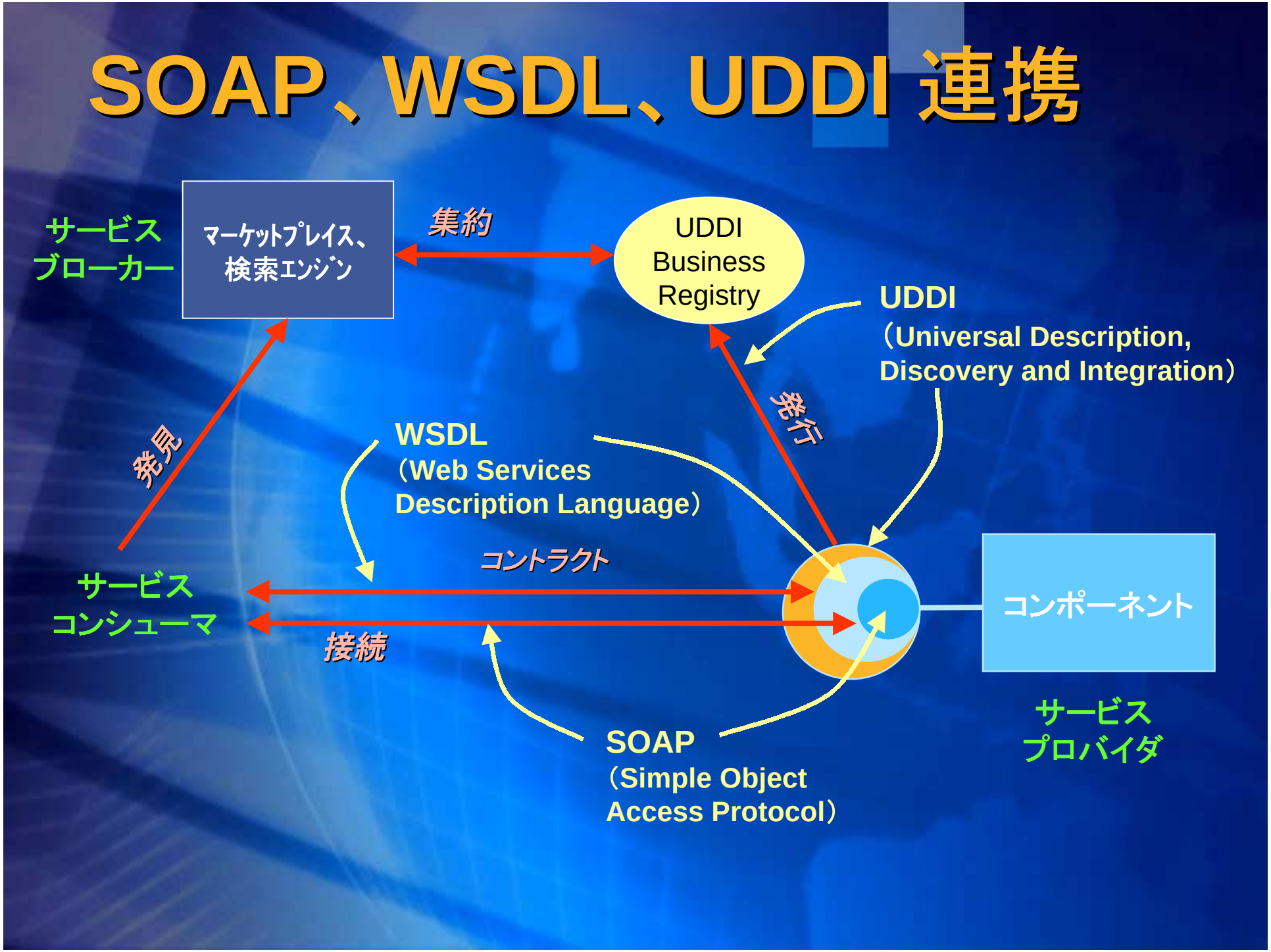


密結合サービスの疎な連携



Source: CBDi Forum

SOAP、WSDL、UDDI 連携



SOAP のバージョンモデル

- 伝統的なメジャー/マイナー番号による管理ではない
- ネームスペースとエレメントで管理
  - Envelop ネームスペース URI – メジャーバージョン
  - Header と Body エレメント – マイナーバージョン
- ただし、SOAP 1.0 への後方互換性は必要

非XMLデータの添付

- Word 文書、画像データ、暗号化データ、など
- MIME multipart/related 構造によるアタッチを仕様化
  - http://msdn.microsoft.com/xml/general/soapattachspec.asp

SOAP Toolkit V2 概要

- SOAPを利用したWebサービスの構築、配布、利用を容易にするツールキット
  - MSXML パーサーチームによる開発（v1はMSDNサンプルチーム）
  - 正式な製品サポートを予定→サードパーティによる対応製品の促進
- COM をプログラミングモデルとして採用
  - Visual Studio 6.0 が利用可能
- 既存アプリケーションのWebサービス化が今日から始められる

SOAP Toolkit V2 の利点

- XML、SOAPの深い知識の必要なし
- ワイアープロトコルを直接扱う必要なし
- ファイアーウォール問題の軽減
- 実装のし易さ(COMプログラミングモデル)
- 配布のし易さ(コンポーネントダウンロードなし)
- 既存のツールセット、スキルの活用

SOAP Toolkit V2 の内容物

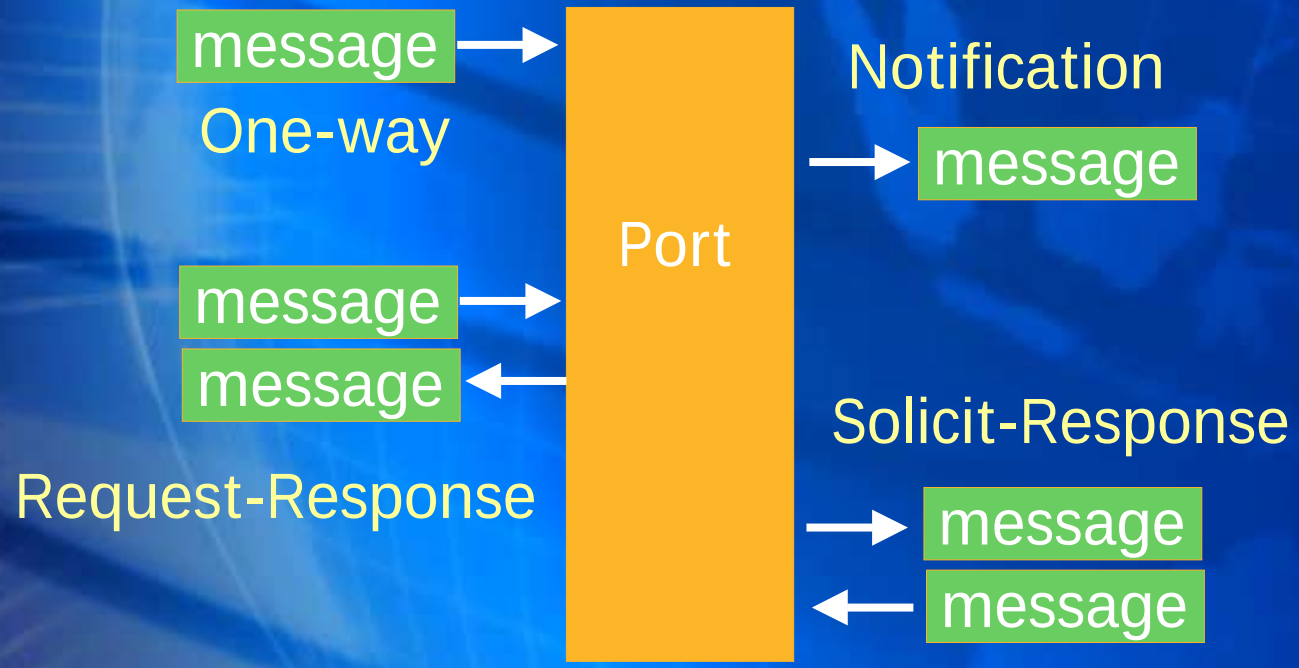
- SOAPをCOMより利用するためのコンポーネント群
  - WSDL で記述されたWebサービスを呼び出すクライアントコンポーネント
  - クライアントからのSOAP要求をCOMコンポーネントにディスパッチさせるサーバーコンポーネント
  - 文書スタイルのSOAPメッセージから“エンティティ”コンポーネントを抽出するジェネレータ
  - SOAPメッセージをマーシャリング、転送、アンマーシャリングするトランスポートコンポーネント
- COM コンポーネントからタイプ情報を取り出し、コントラクト、バインド情報を生成するジェネレータ
- プログラミングガイド、リファレンス、サンプルなど

WSDL：Web サービスの記述

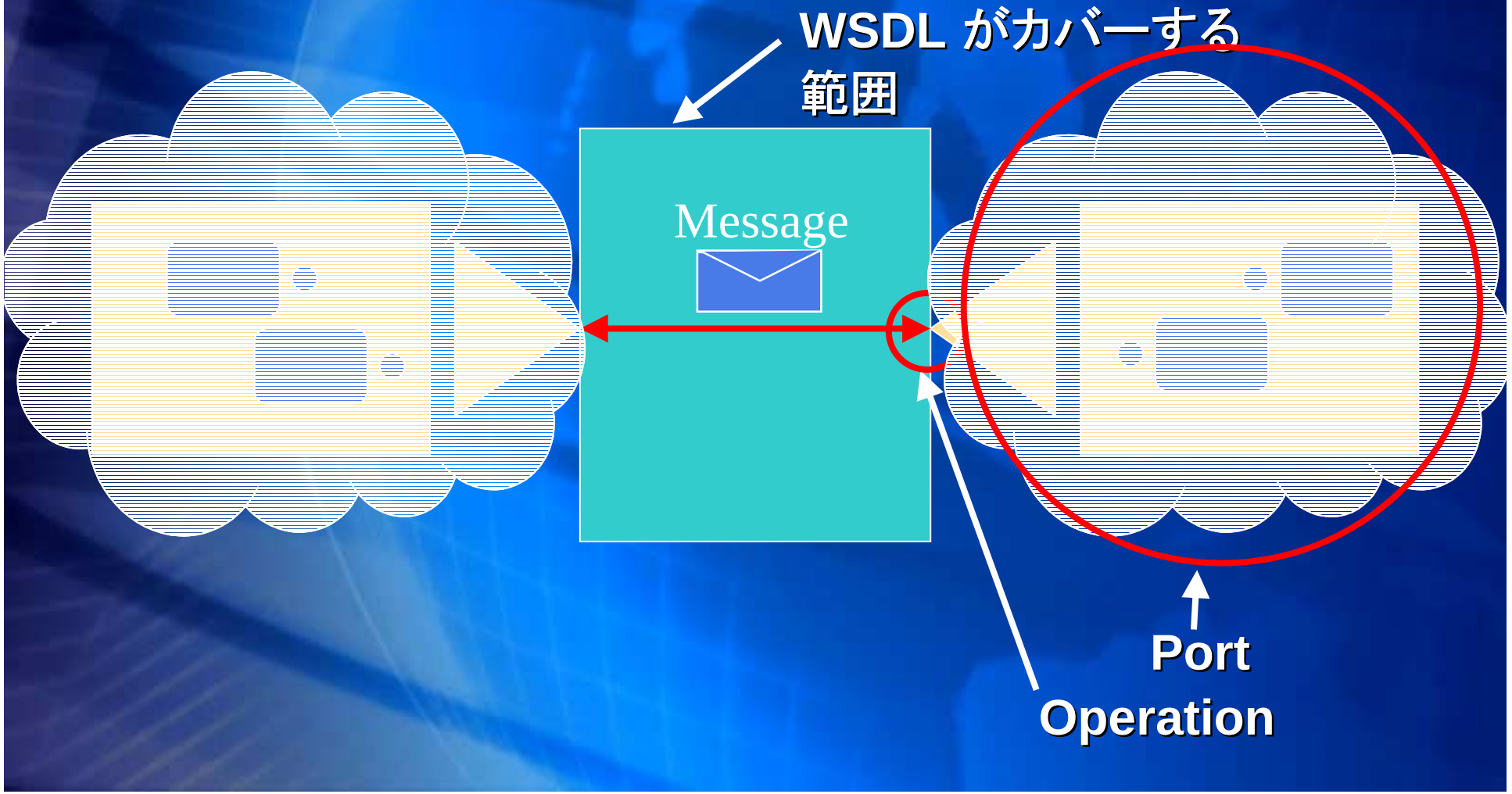
- *Web Services Description Language*
- サービス記述のためのXMLベースの文法
  - システム間通信のための「契約」
  - 他社システムが自社システムと通信する方法を理解
- NASSL（IBM）、SCL、SDL（共にMS）を統合
- メッセージ交換を行う、複数のエンドポイントを記述する
  - RPC、文書スタイルともに可能
- WSDL 1.0 仕様を公開
  - http://msdn.microsoft.com/xml/general/wsdl.asp
  - http://www.microsoft.com/japan/developer/workshop/xml/general/wsdl.asp（日本語訳）

基本概念

- Port – エンドポイント
- Message – エンドポイント間を流れるデータ
- Operation – メッセージのコンビネーション

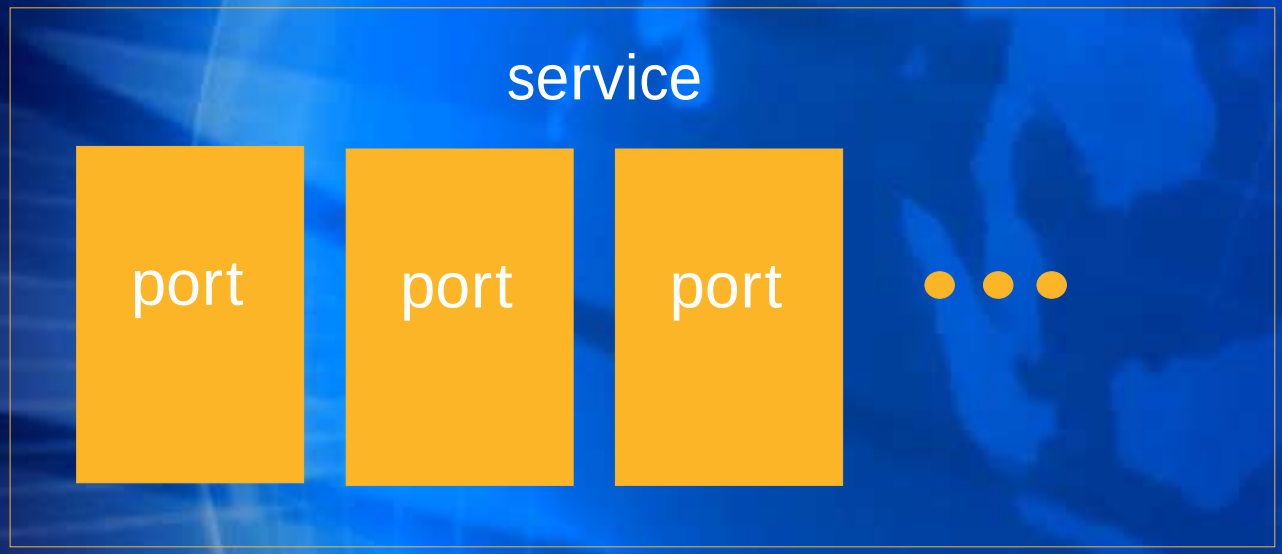


WSDL とサービス連携



サービス

- Service – 関連するPortのコレクション



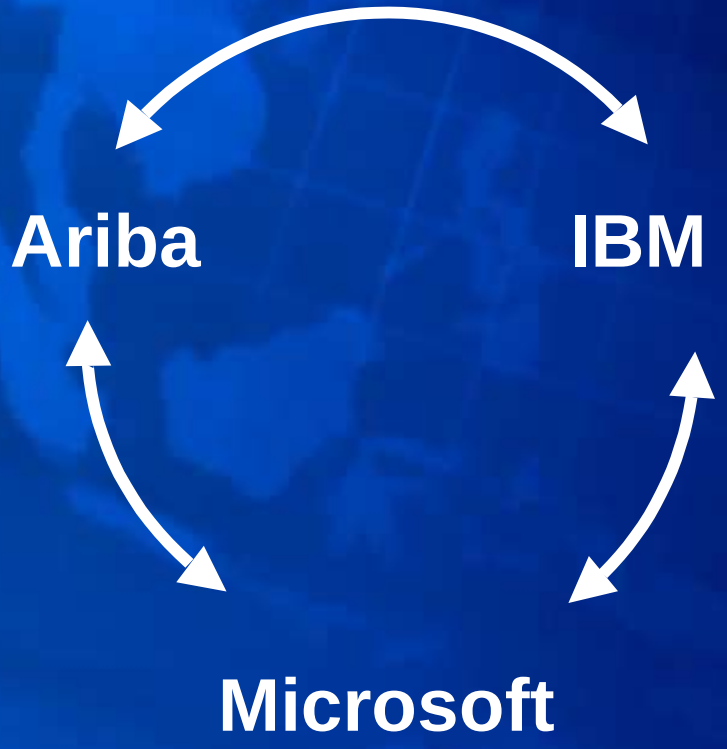
オーケストレーション、コンポジションは（まだ）含まれず

再利用可能

- PortType
  - オペレーションとメッセージのセット
  - 抽象的
    - プロトコルやワイアー形式の情報は含まれず
- Binding
  - PortType をプロトコルやワイアー形式に結びつける
  - SOAP 1.1?HTTP Get/Post?MIME
- プロトコルや距離などの条件で適切なポートを選択可能
  - 例: 本屋の PortType
  - 2つの Binding:
    - SOAP over HTTP
    - SOAP over SMTP

UDDI とは？

- Webサービスの相互運用性と適用を早めるためのプロジェクト
  - サービスの記述と発見のための標準仕様
  - Web上のビジネスレジストリの共有オペレーション
- 業界とビジネスリーダー間のパートナーシップ
- ***Universal Description, Discovery*, and *Integration***



どのようなソリューションを提供するか？

B2Bの拡大 → 中規模の製造業者は、それぞれが独自の標準とプロトコルを採用する400の顧客とオンライン上の関係を作る必要がある

検索能力の向上 → オーストラリアの花屋が、世界中のマーケットプレイスに“出店”したいが、その方法がわからない

サービス集約の容易化 → B2B マーケットプレイスは各業界の関連サプライヤからカタログデータを入手し、それに加え配送業者や保険会社への接続も必要である

サービスの記述

サービスの発見

サービスの統合

UDDI v1 の動作

1. 
ソフトウェア会社、標準化団体、開発者などが様々なサービスのタイプをレジストリに登録

2. 
企業は自らがサポートするサービスの記述をレジストリに登録

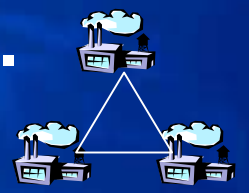
UDDI ビジネスレジストリ（UBR）

ビジネス
レジストリ

サービスタイプ
レジストリ

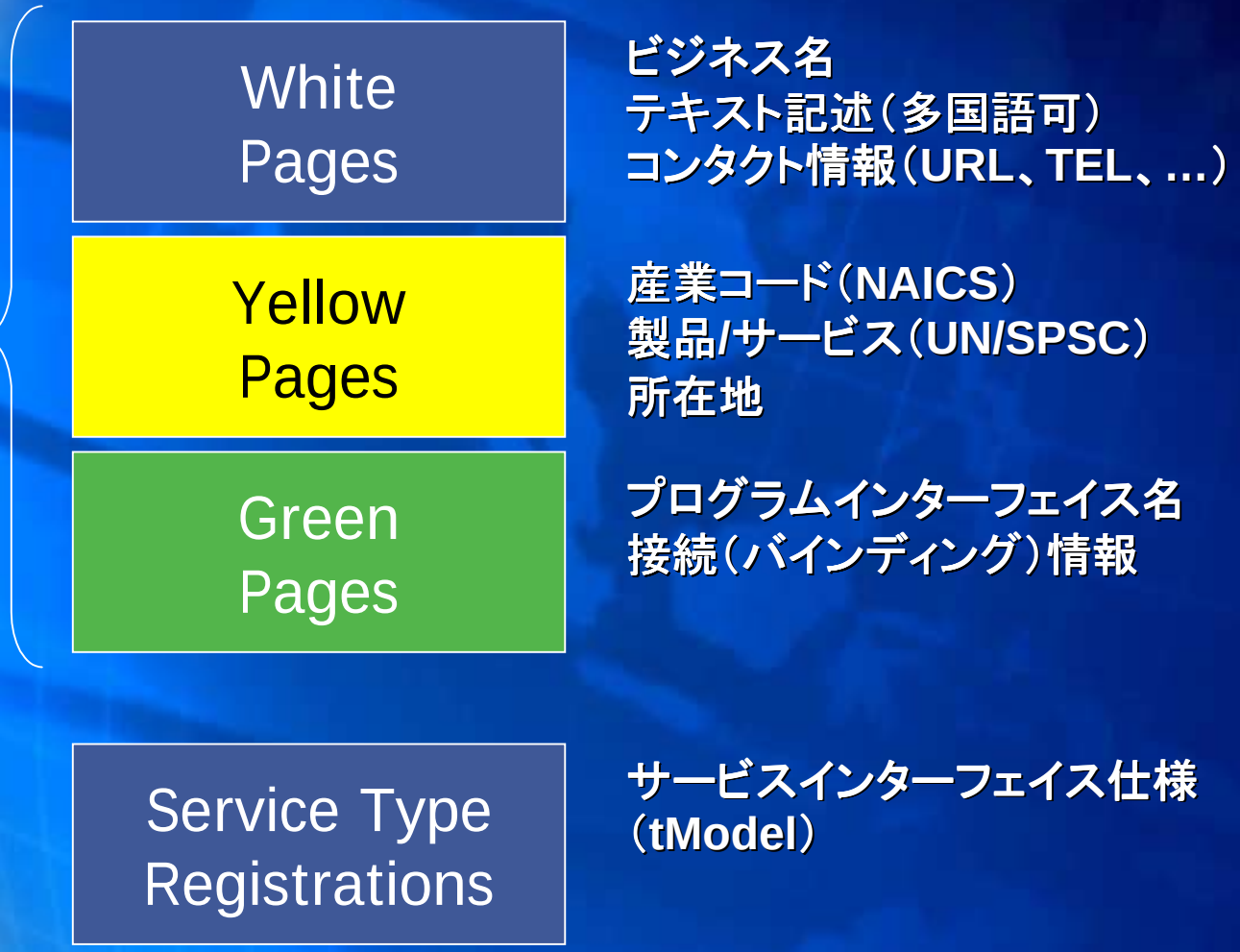
3. UBR は各サービスとビジネスにユニークな識別子をふる

4. 
マーケットプレイス、検索エンジン、ビジネスアプリなどが他の組織が提供するサービスを発見するためにレジストリに問い合わせをする

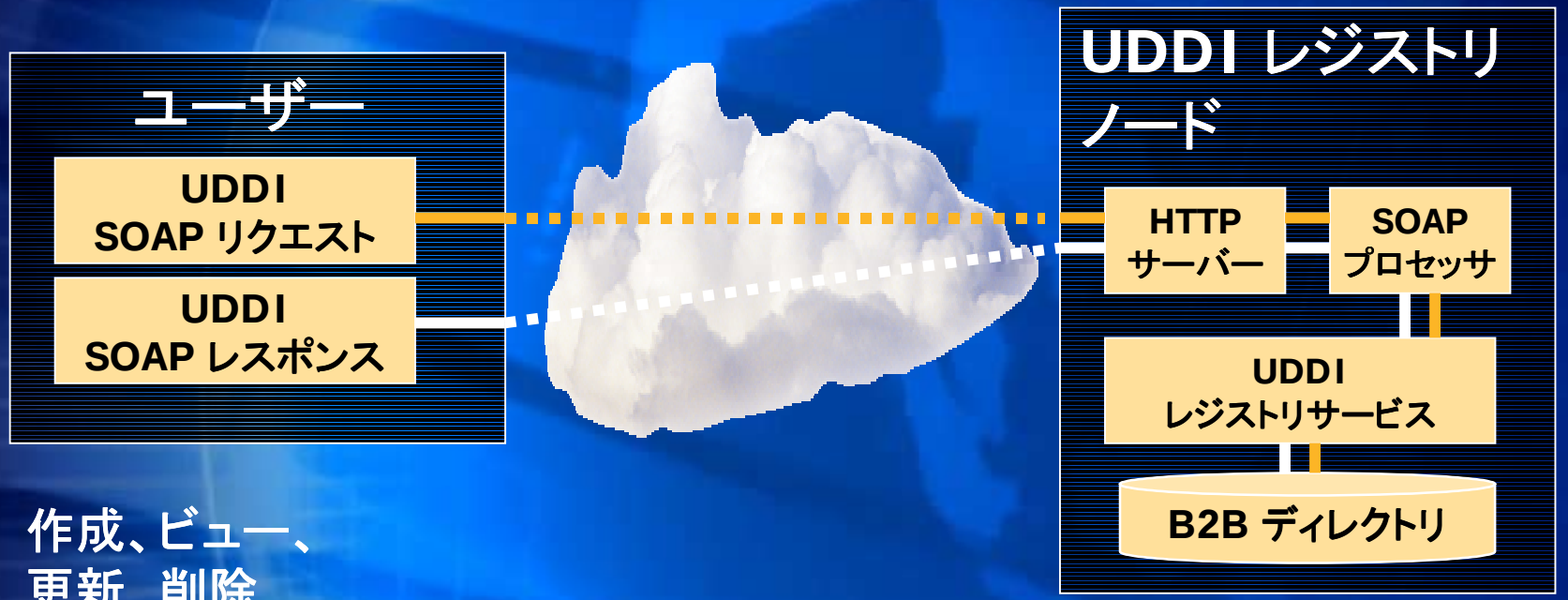
5. 
企業はWebをまたがる統合を促進するためにこれらのデータを利用する

レジストリのデータ

- 企業は自社に関する公開情報を登録する
- 標準化団体、開発者、企業などは各々のサービスタイプの情報を登録する



UDDI と SOAP



作成、ビュー、
更新、削除

実装中立

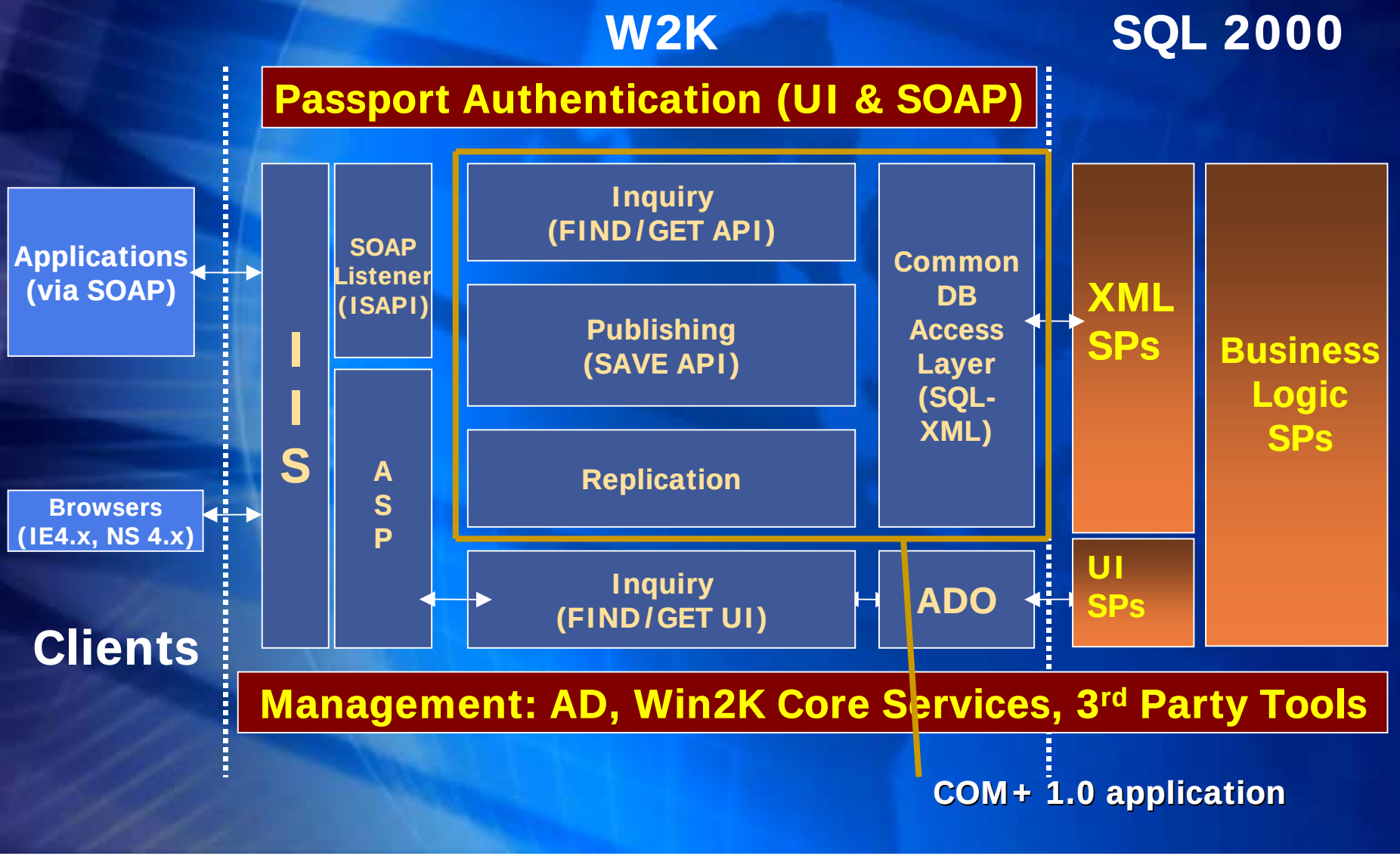
レジストリ API

- 問い合わせ用 API
  - 検索
    - find_business
    - find_service
    - find_binding
    - find_tModel
  - 詳細情報の取得
    - get_businessDetail
    - get_serviceDetail
    - get_bindingDetail
    - get_tModelDetail
- 発行用 API
  - セーブ
    - save_business
    - save_service
    - save_binding
    - save_tModel
  - 削除
    - delete_business
    - delete_service
    - delete_binding
    - delete_tModel
  - セキュリティ…
    - get_authToken
    - discard_authToken

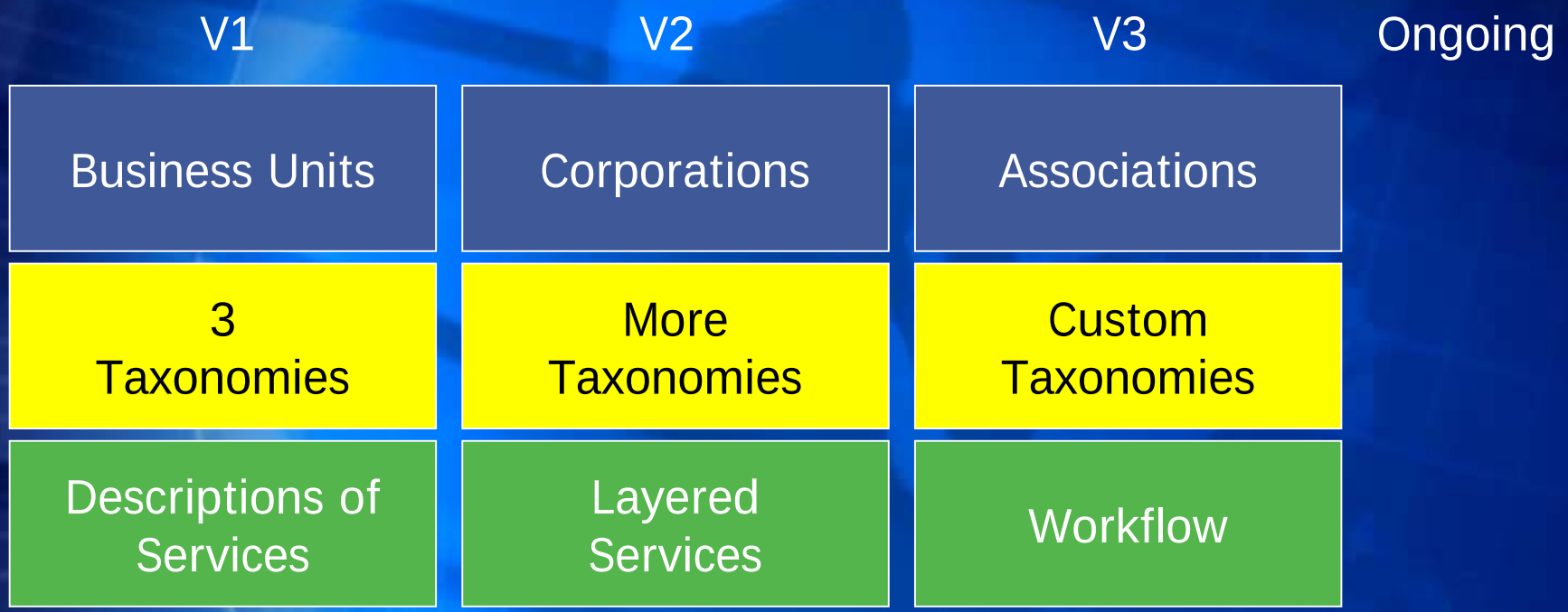
UDDI レジストリの操作

- Microsoft UDDI SDK（ベータ）
  - Visual Studio 6.0対応
  - UDDI APIをCOMでラッピング
  - コンポーネント、プログラマーズガイド、サンプルプログラムなど

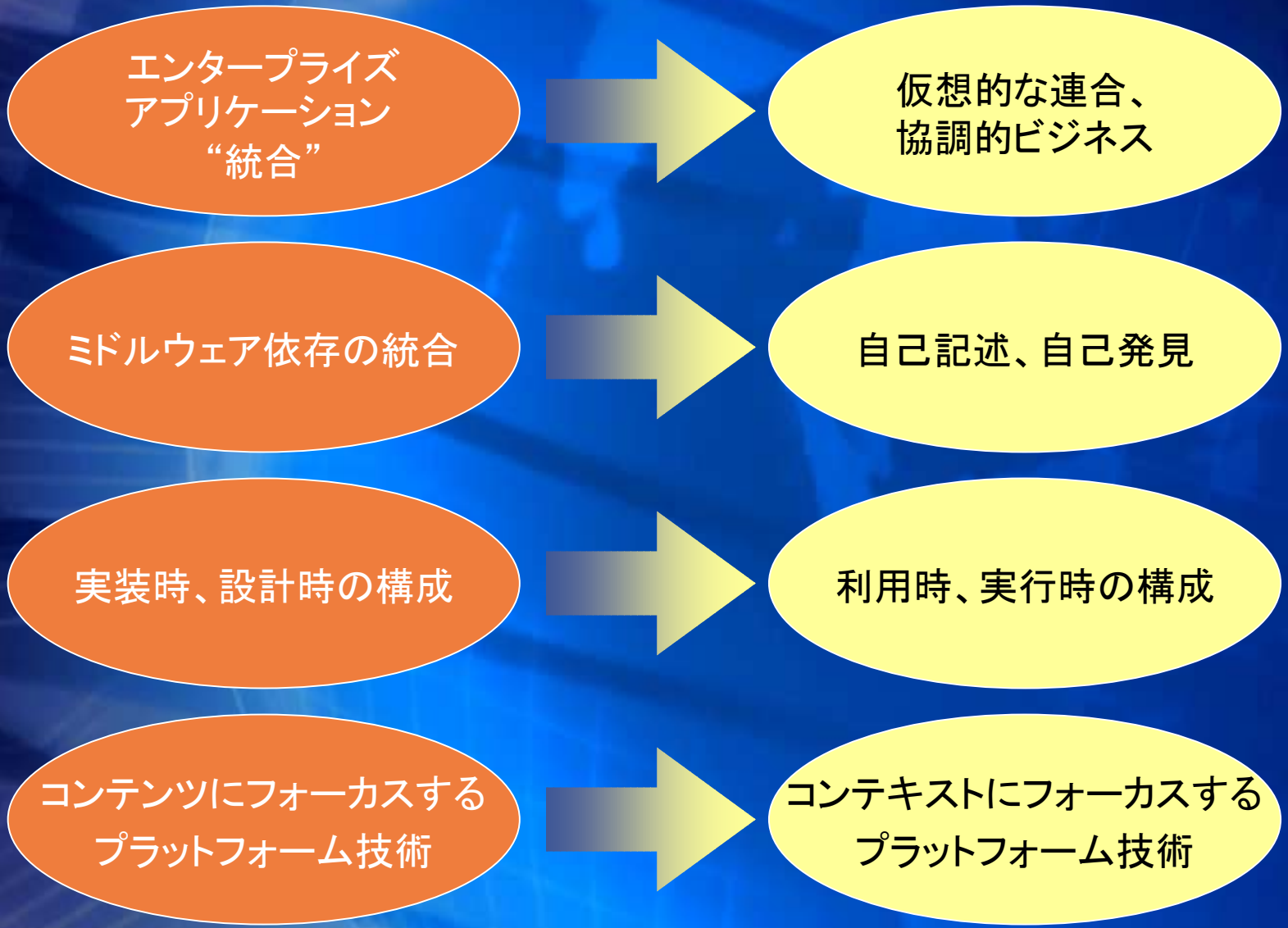
UDDI.MICROSOFT.COM



UDDI ロードマップ


V1
V2
V3
Ongoing
Business Units
Corporations
Associations
3 Taxonomies
More Taxonomies
Custom Taxonomies
Descriptions of Services
Layered Services
Workflow
Standards Body

Web サービス時代の開発



まとめ

- Web サービスは顧客、パートナーなどとの電子的関係をJust-in-Timeで結ぶ
- Web サービスは密結合サービスの疎結合な連携を実現する
  - .NET は COM(+) と XML の統合
- Web サービスはプラットフォーム非依存なメッセージ交換インフラを活用する
  - SOAP、WSDL、UDDI、…