

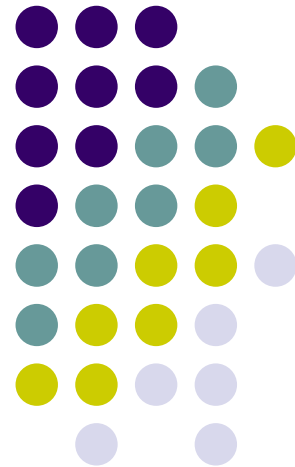
# 次世代グリッド基盤OGSA におけるC言語動作環境 提供システムの試作と評価

濱野 智行<sup>\*1</sup>, 中田 秀基<sup>\*2\*1</sup>,  
鈴村 豊太郎<sup>\*1</sup>, 松岡 聡<sup>\*1\*3</sup>

<sup>\*1</sup> 東京工業大学

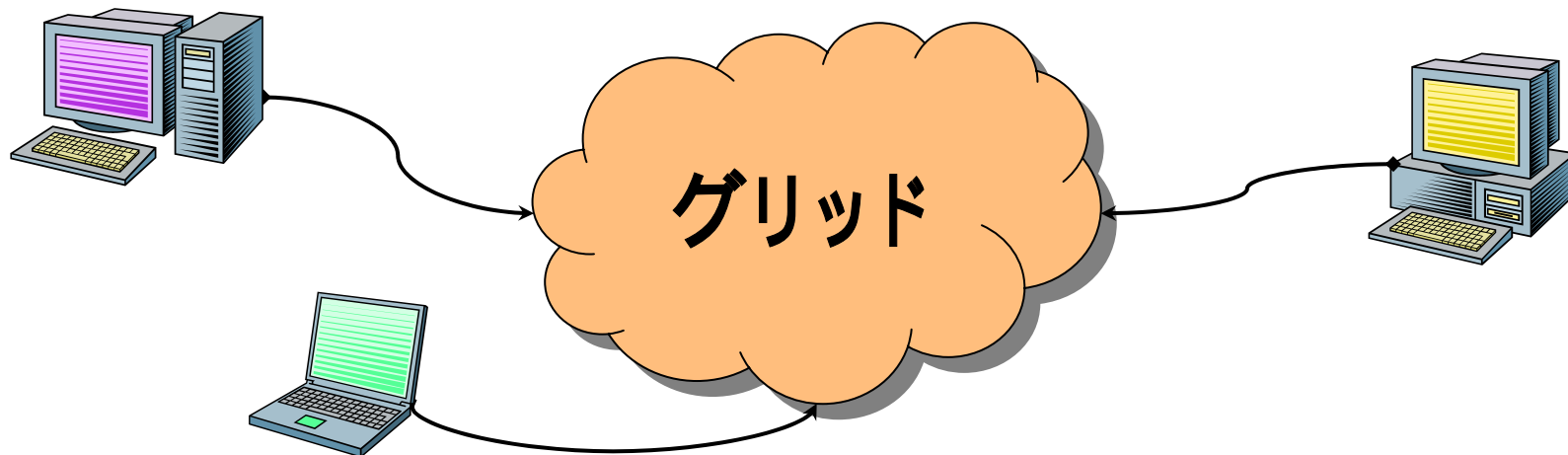
<sup>\*2</sup> 産業技術総合研究所

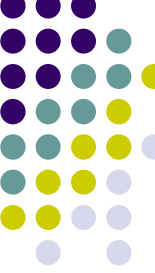
<sup>\*3</sup> 国立情報学研究所



# 本研究の背景

- グリッドコンピューティング
  - 広域ネットワーク上に分散した計算資源を活用
  - Globus Toolkit : 実装の1つ
  - 新しいアーキテクチャOGSAの登場





# Open Grid Services Architecture(OGSA)

- 従来のグリッド技術とWebサービスとの融合
  - XMLベースプロトコルにより高い相互運用性を確保
    - WSDL : Webサービスのインタフェースを定義
    - SOAP : Webサービスの通信プロトコル
  - 基盤部分はOGSIで規定
  - GT3(Globus Toolkit 3.0) : OGSA実装の1つ

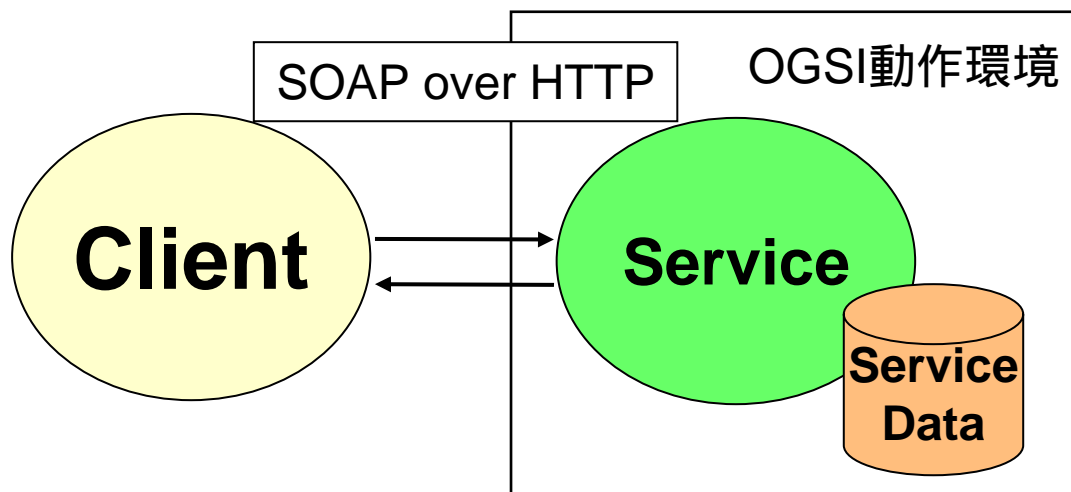
# Open Grid Services Infrastructure(OGSI)

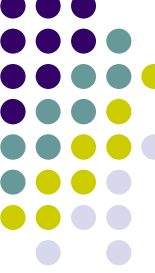


- Webサービスに状態を持たせたコンポーネントモデル
  - サービスの特徴や挙動を定義
    - インタフェイス
    - インスタンス生成機構
    - 生存期間管理
    - 通知
    - etc...
  - サービスデータ：サービスに状態を持たせる機構

# OGSIコンポーネント

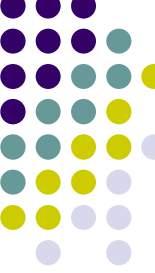
- クライアントからサービスへのRPC
  - 通信プロトコルにSOAP over HTTP
- サービスはOGSI動作環境に配置
  - サービス内にサービスデータを保持





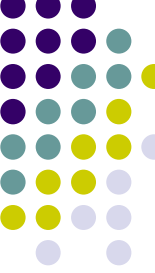
# OGSIの問題点

- XMLベースプロトコル通信による性能低下
  - データサイズの肥大化
  - パーシングコスト大
- 現状のOGSI実装ではJava以外の動作環境が未提供
  - C/C++/Fortranによるレガシーアプリケーションがサービスできない



# 本研究の目的

- OGSi実装(GT3)上にC言語動作環境を提供
  - GridRPCシステムNinf-1を用いてC言語動作環境を実現
    - C/C++/Fortranで記述されたサービスが配置可能に
  - サービス開発補助ツールの提供
    - サービスのスケルトンを自動生成
    - サービスデータを透過的に利用可能
  - 構築システムの通信性能評価
    - 構築システムとGT3 Java環境との通信コストを計測、比較評価



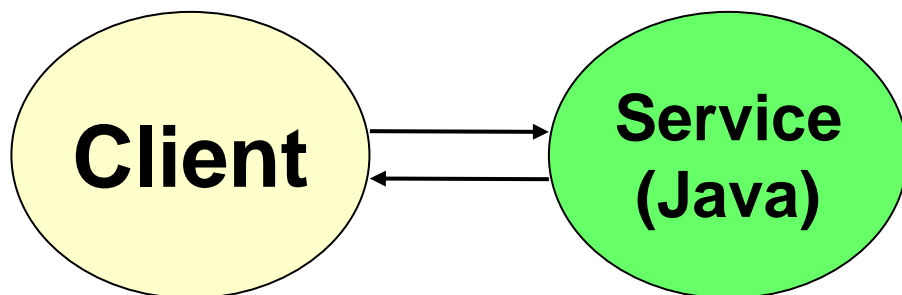
# 発表の概要

- OGSiの概要
- GridRPCシステムNinf-1の概要
- OGSiにおけるC言語動作環境提供システムの提案
  - 提案システムの概要
  - Ninf-1 on OGSi
  - Ninf-1サービススケルトンジェネレータ
- 提案システムの評価



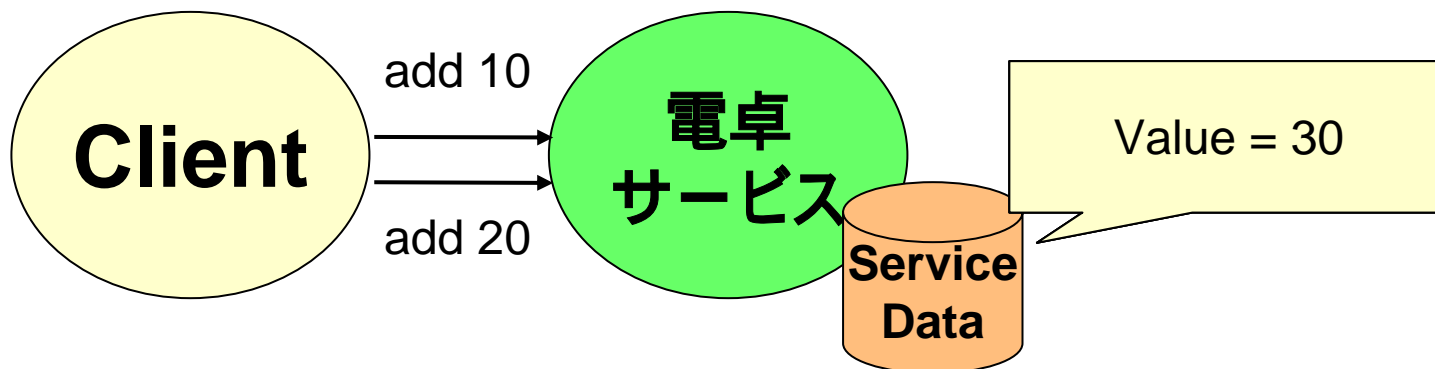
# OGSIコンポーネントモデル

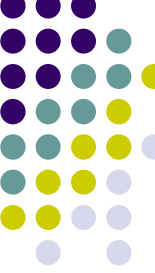
- Webサービスと類似のコンポーネントモデル
  - クライアントとサービスが通信
- 現在のOGSI実装では、サービスはJava以外で記述不可能



# サービスデータ

- サービスの状態を保持する機構
  - 外部からサービスの状態参照が可能
- サービスが処理に利用するテンポラリバッファ的な利用も可能
  - 容易に参照・更新できるように、処理を行うサービスがサービスデータを保持すべき



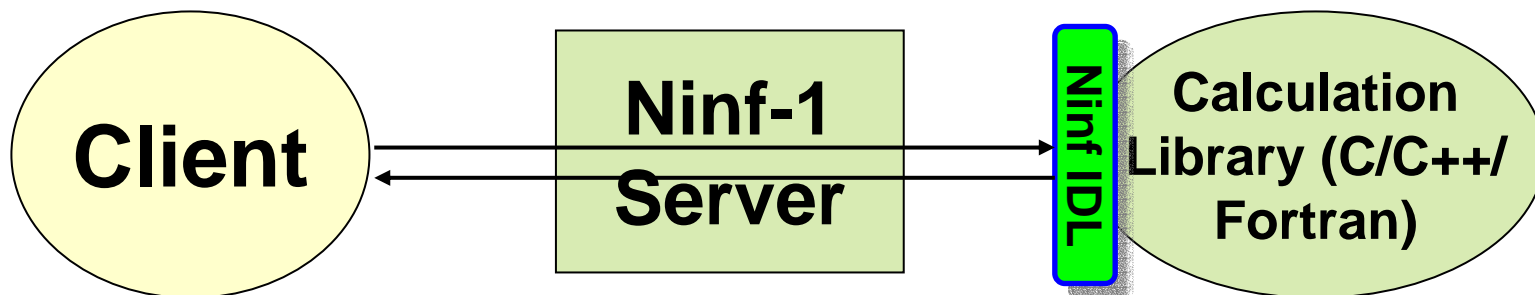


# 発表の概要

- OGSiの概要
- GridRPCシステムNinf-1の概要
- OGSiにおけるC言語動作環境提供システムの提案
  - 提案システムの概要
  - Ninf-1 on OGSi
  - Ninf-1サービススケルトンジェネレータ
- 提案システムの評価

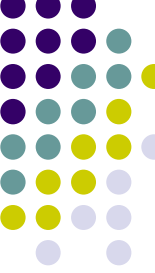
# GridRPCシステムNinf-1

- グリッド上のRPCシステム
  - クライアントがNinf-1サーバを介して計算ライブラリを呼び出す
- 計算ライブラリのインタフェースをNinf IDLで記述



クライアント例

```
NinfClient client = new NinfClient("add");  
client.call(args);
```



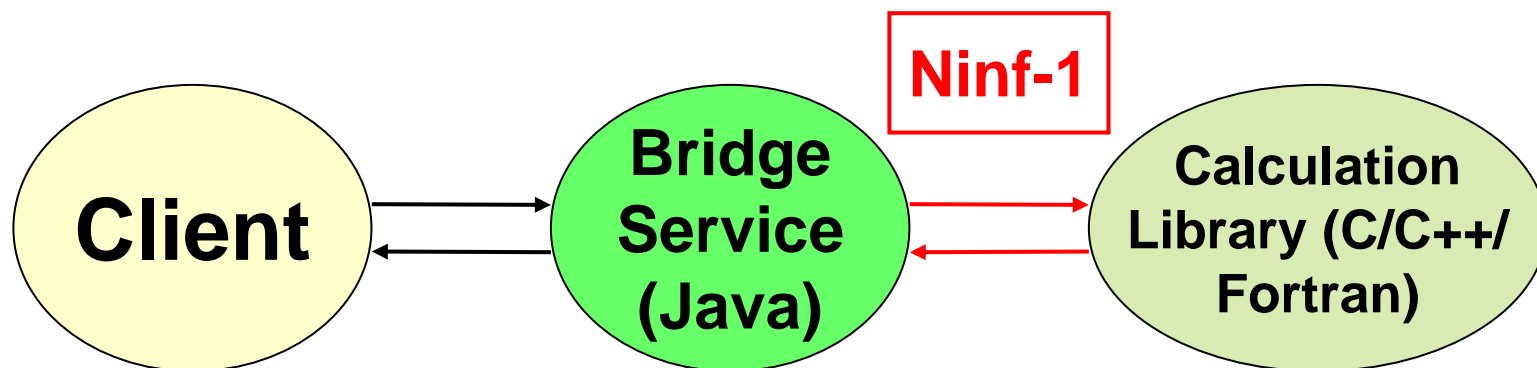
# 発表の概要

- OGSiの概要
- GridRPCシステムNinf-1の概要
- OGSiにおけるC言語動作環境提供システムの提案
  - 提案システムの概要
  - Ninf-1 on OGSi
  - Ninf-1サービススケルトンジェネレータ
- 提案システムの評価

# OGSIにおけるC言語動作環境提供システムの概要

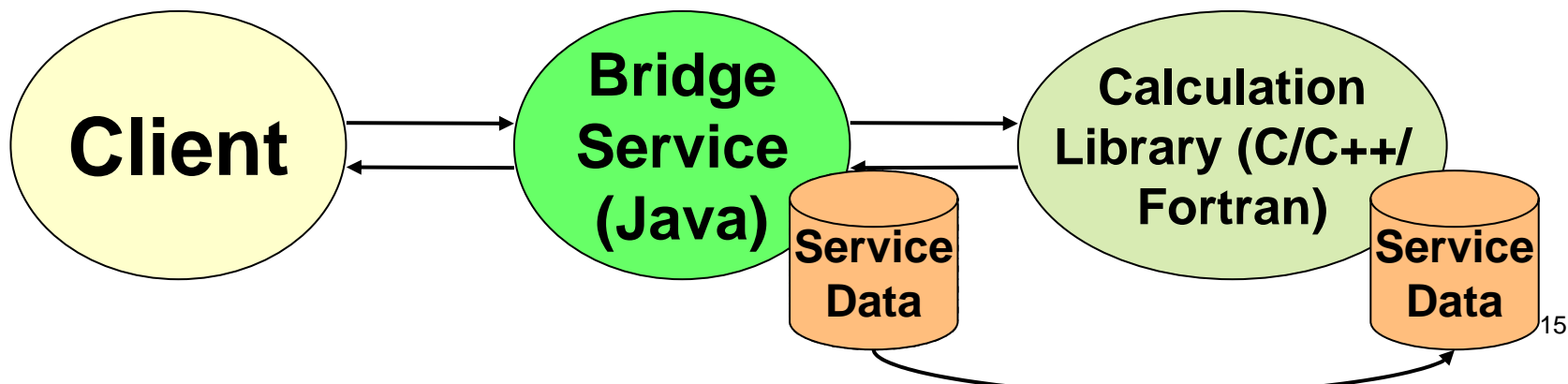


- クライアントのリクエストをブリッジサービスが計算ライブラリにブリッジ
  - 実際にリクエスト进行处理するのは計算ライブラリ
- GridRPCシステムNinf-1をローカルに用いてJavaからC/C++/Fortranへのブリッジを実現



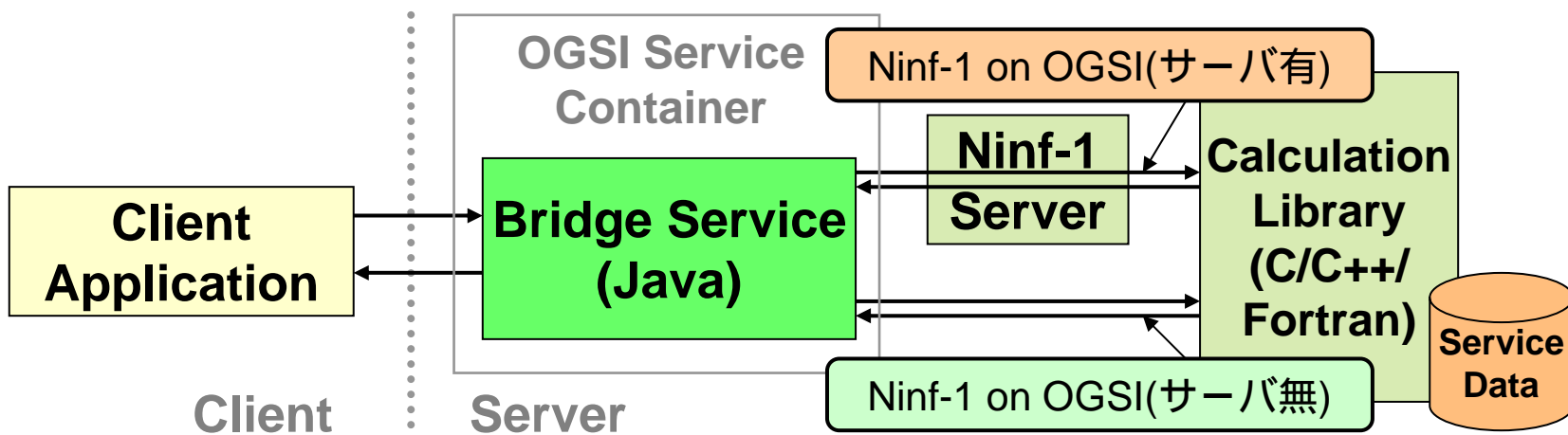
# サービスデータのブリッジ

- 提案システムにおいてサービスデータは、リクエストの処理を行う計算ライブラリに保持
  - リクエストと同様にサービスデータをブリッジ
  - クライアントはブリッジを意識することなくサービスデータを利用したい



# Ninf-1 on OGSi

- GT3上に提案システムを実装
  - C/C++/Fortranで記述されたサービスを配置可能
  - Ninf-1サーバを使用しないブリッジの設定が可能
    - ブリッジサービスと計算ライブラリとが同一ホスト内

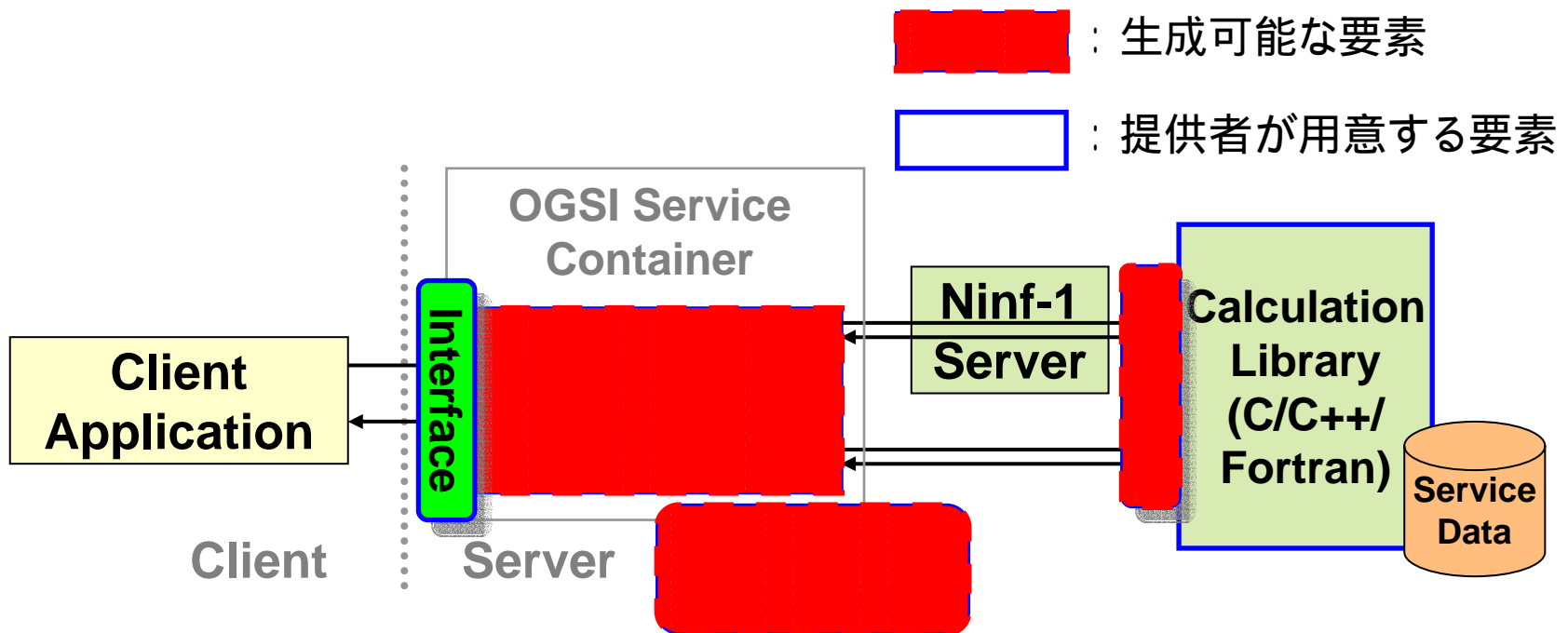


ブリッジを追加したことで開発コストが増大



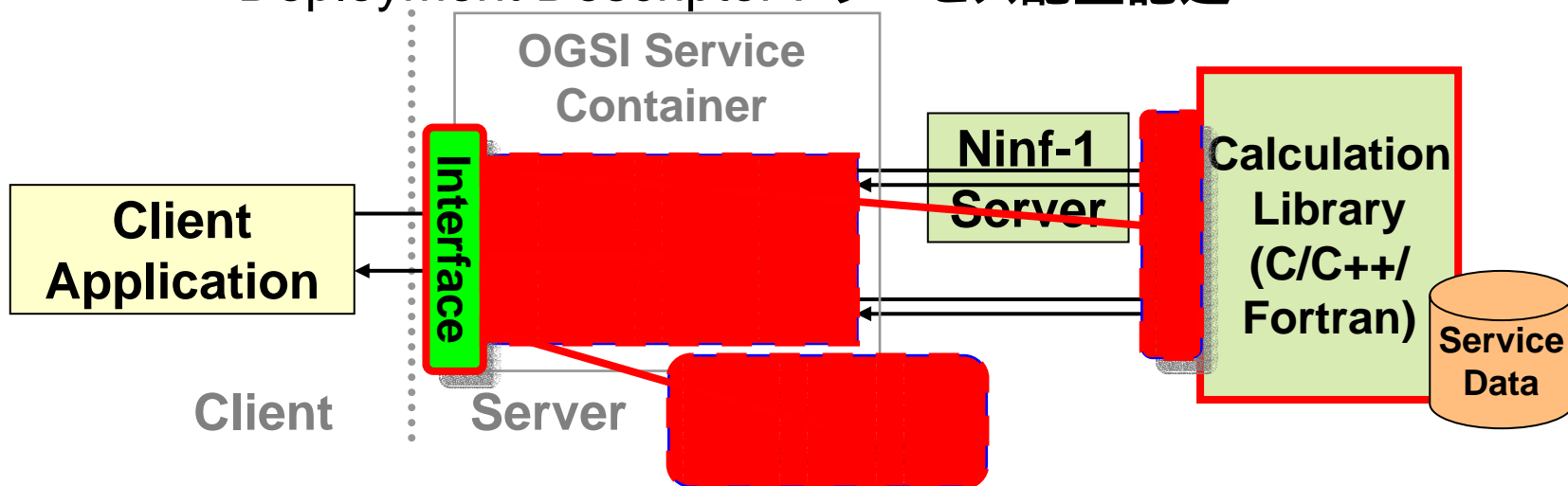
# 開発コストが増大する要因

- Ninf-1 on OGSiでは用意する要素が増加
- いくつかの要素は他要素の情報から生成可能



# Ninf-1サービススケルトンジェネレータ

- サービス開発が容易になる開発補助ツール
  - ブリッジサービスのJavaインタフェースからサービスのスケルトンを生成
    - ブリッジサービス
    - Ninf IDL : 計算ライブラリのインタフェース記述
    - Deployment Descriptor : サービス配置記述



# Ninf-1サービススケルトンジェネレータ

## - スケルトン生成例 -



入力

Javaインタフェース

```
package jp.ac.matsulab.lindyna.impl;
public interface Counter {
    public int add(int val);
    public int subtract(int val);
    public void setState(int val);
    public int getState();
}
```

自動生成

ブリッジサービス

```
public class C
    implements
    private static
    private Grp
    public Coun
    super("Cou
    getHandle
```

Deployment Descriptor

```
<?xml version
<deployment
name="defau
wsdd/"
xmlns:java="
<service nam
provider="Ha
```

Ninf IDL

```
Module lindyna;
DefClass Counter
{
    DefState {
        int Value;
    }
    Define getState(OUT int arg0[1])
```

出力

# Ninf-1サービススケルトンジェネレータ

## - ブリッジサービスの生成 (1/2) -



- Ninf-1を用いたブリッジに必要な処理を行う部分を自動生成
  - 引数配列サイズの指定
    - クライアントアプリケーションから受け取った引数に配列サイズ情報を付加してブリッジ
  - 戻り値配列サイズの指定
    - 引数に戻り値と等しい次元数の配列があったらそのサイズを戻り値のサイズとする
    - 引数に戻り値と等しい次元数の配列がなかったらサイズを空欄にしておき、サービス提供者に手で埋めてもらう

# Ninf-1サービスケルトンジェネレータ

## - ブリッジサービスの生成 (2/2) -



- 次元数の等しい引数有り

- 次元数の等しい引数無し

```
public int[] mmul(int[] arg0, int[] arg1) {  
    int[] ReturnValue = new int[arg0.length];  
    Vector args = new Vector();  
    args.add(new Integer(arg0.length));  
    args.add(arg0);  
    args.add(new Integer(arg1.length));  
    args.add(arg1);  
    args.add(new Integer(ReturnValue.length));  
    args.add(ReturnValue);  
    callRemoteClient("myService/mmul", args);  
  
    return ReturnValue;  
}
```

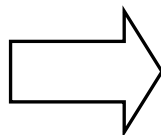
```
public int[][] vmul(int[] arg0, int[] arg1) {  
    int[][] ReturnValue = new int[][] /** size **/  
    Vector args = new Vector();  
    args.add(new Integer(arg0.length));  
    args.add(arg0);  
    args.add(new Integer(arg1.length));  
    args.add(arg1);  
    args.add(new Integer(ReturnValue.length));  
    args.add(new Integer(ReturnValue[0].length));  
    args.add(ReturnValue);  
    callRemoteClient("myService/vmul", args);  
  
    return ReturnValue;  
}
```

# Ninf-1サービススケルトンジェネレータ - 透過的サービスデータ利用の実現 -



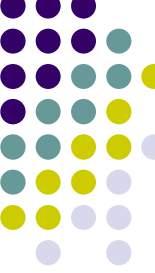
- setAAA(), getAAA() という名前のメソッドを追加
  - AAAというサービスデータを利用可能なスケルトンを生成

```
package jp.ac.matsulab.lindyna.impl;  
public interface Counter {  
    public int add(int val);  
    public int subtract(int val);  
    public void setState(int val);  
    public int getState();  
}
```



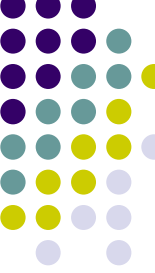
```
/**  
 * getValue  
 * @ogsa:service-data  
 *   name      = "State"  
 *   minOccurs = "1"  
 *   maxOccurs = "1"  
 *   mutability = "mutable"  
 *   modifiable = "true"  
 *   nillable  = "false"  
 */  
public int getValue() throws RemoteException {  
    Vector args = new Vector();  
    int[] ReturnValue = new int[1];  
    args.add(ReturnValue);
```

クライアントはブリッジを意識することなくサービスデータを利用可能



# 発表の概要

- OGSiの概要
- GridRPCシステムNinf-1の概要
- OGSiにおけるC言語動作環境提供システムの提案
  - 提案システムの概要
  - Ninf-1 on OGSi
  - Ninf-1サービススケルトンジェネレータ
- 提案システムの評価

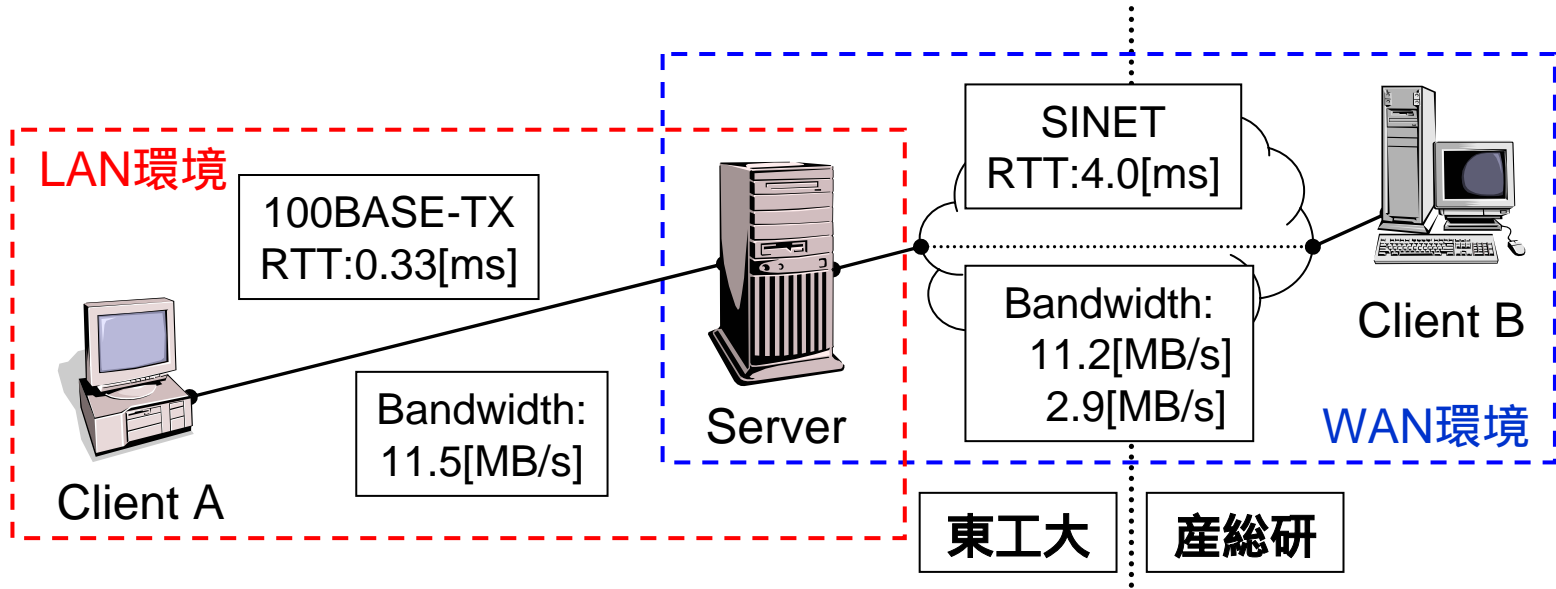


# 評価

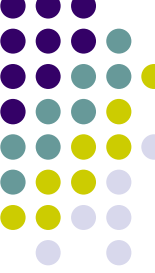
- RPCシステムのping-pongによる比較評価
  - Ninf-1 on OGS1 (Ninf-1サーバ有/無)
  - GT3 Java環境
  - Ninf-1
- サービスデータブリッジの評価
  - サービスデータをブリッジする際のコスト
- Ninf-1サービススケルトンジェネレータの評価
  - サービスの開発工数の定性的評価



# 評価環境



	CPU	メモリ
サーバ	AthlonXP 1.2GHz	1GBytes
クライアントA	Athlon 1.3GHz	1GBytes
クライアントB	PentiumIII 1.4GHz	2GBytes

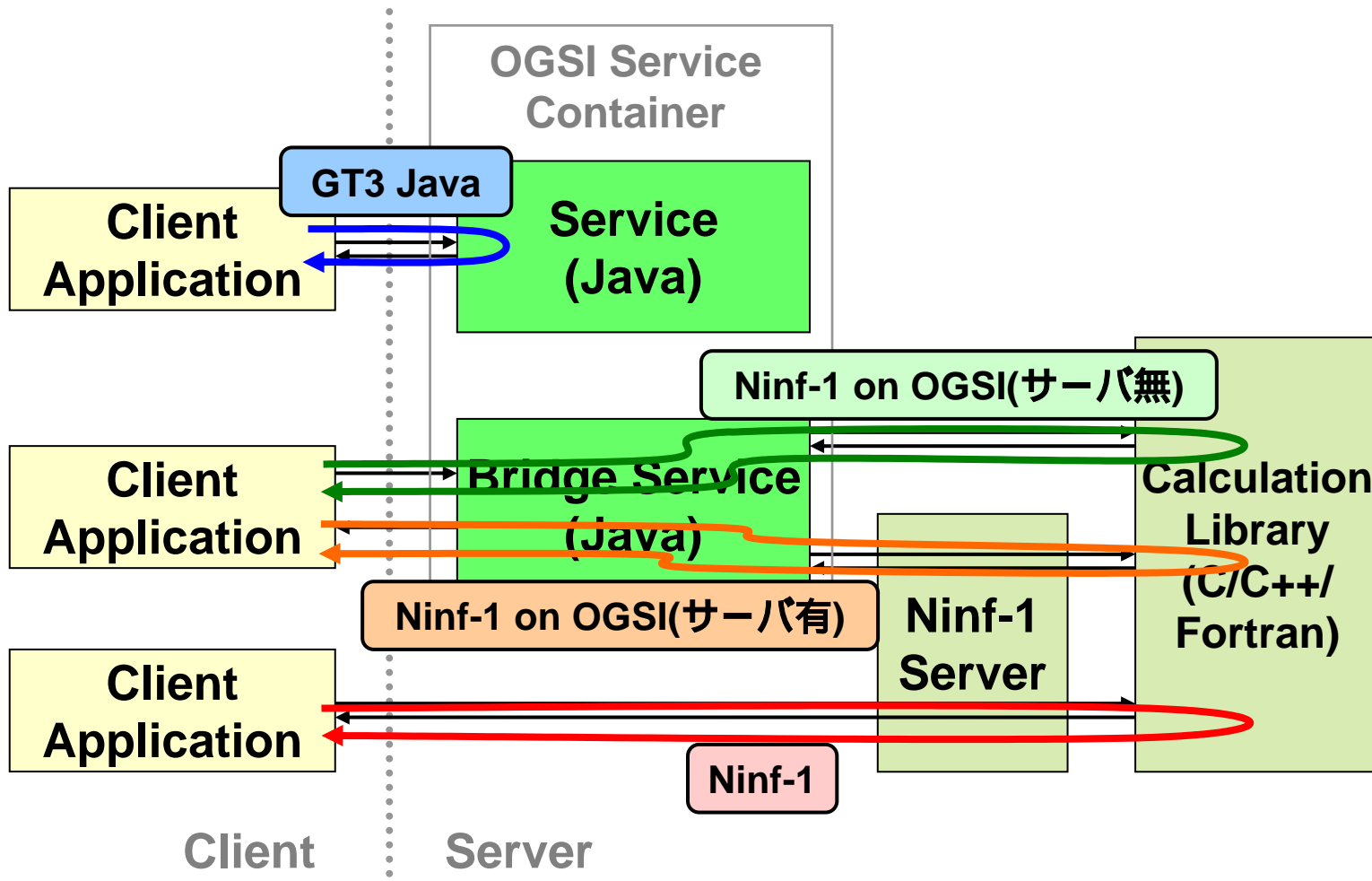


# RPCシステムの比較評価

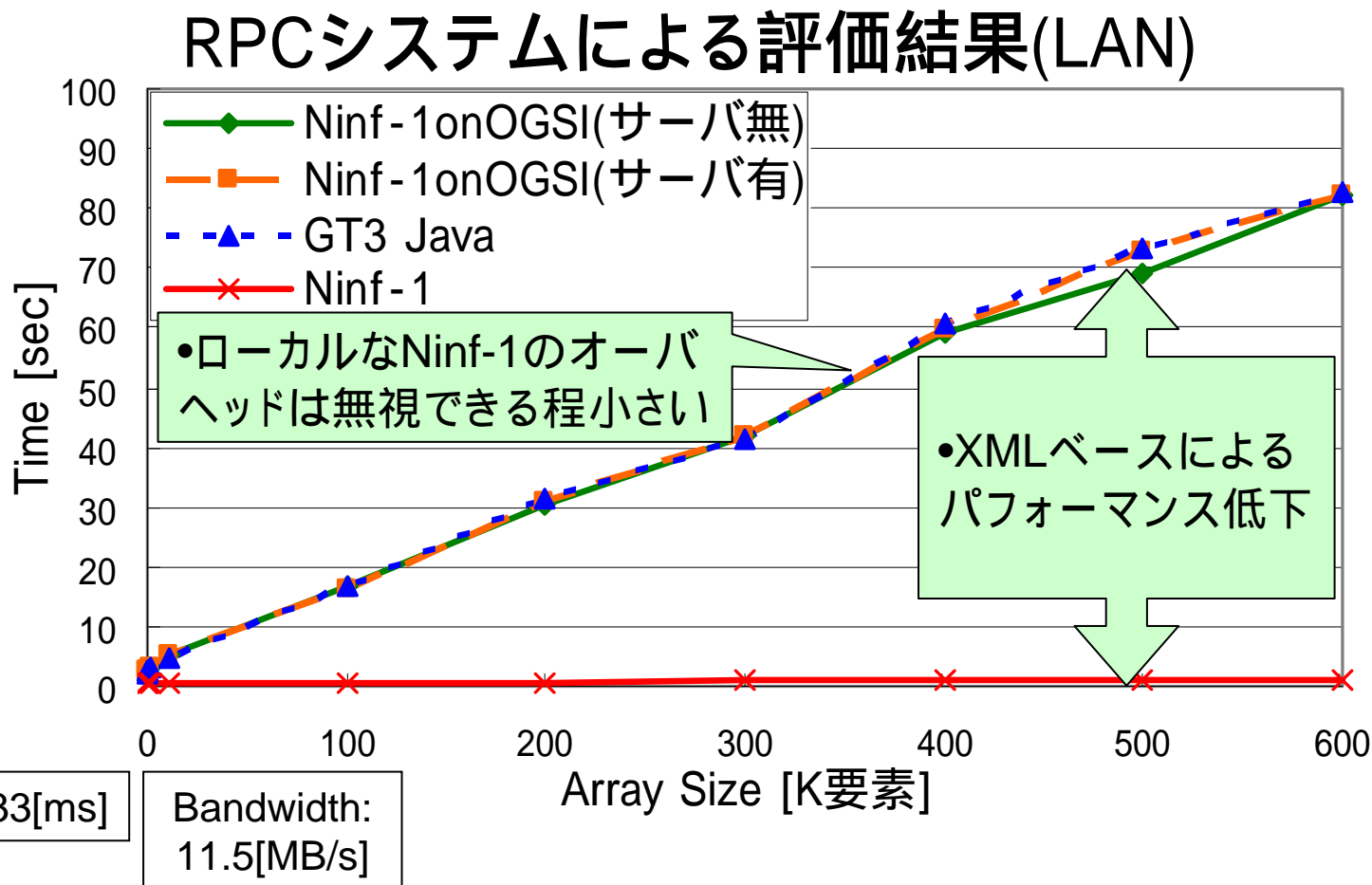
- Ninf-1 on OGSi上にping-pongサービスを配置
  - int型1次元配列をping-pongするサービス
  - LAN/WAN環境でシステム全体の時間コストを計測
- 他のRPCシステムとの比較評価
  - GT3 Java環境でのRPCシステム
  - GridRPCシステムNinf-1

	Ninf-1 on OGSi (サーバ有)	Ninf-1 on OGSi (サーバ無)	GT3 Java	Ninf-1
XMLベース プロトコル通信				×
リクエストの ブリッジ			×	×

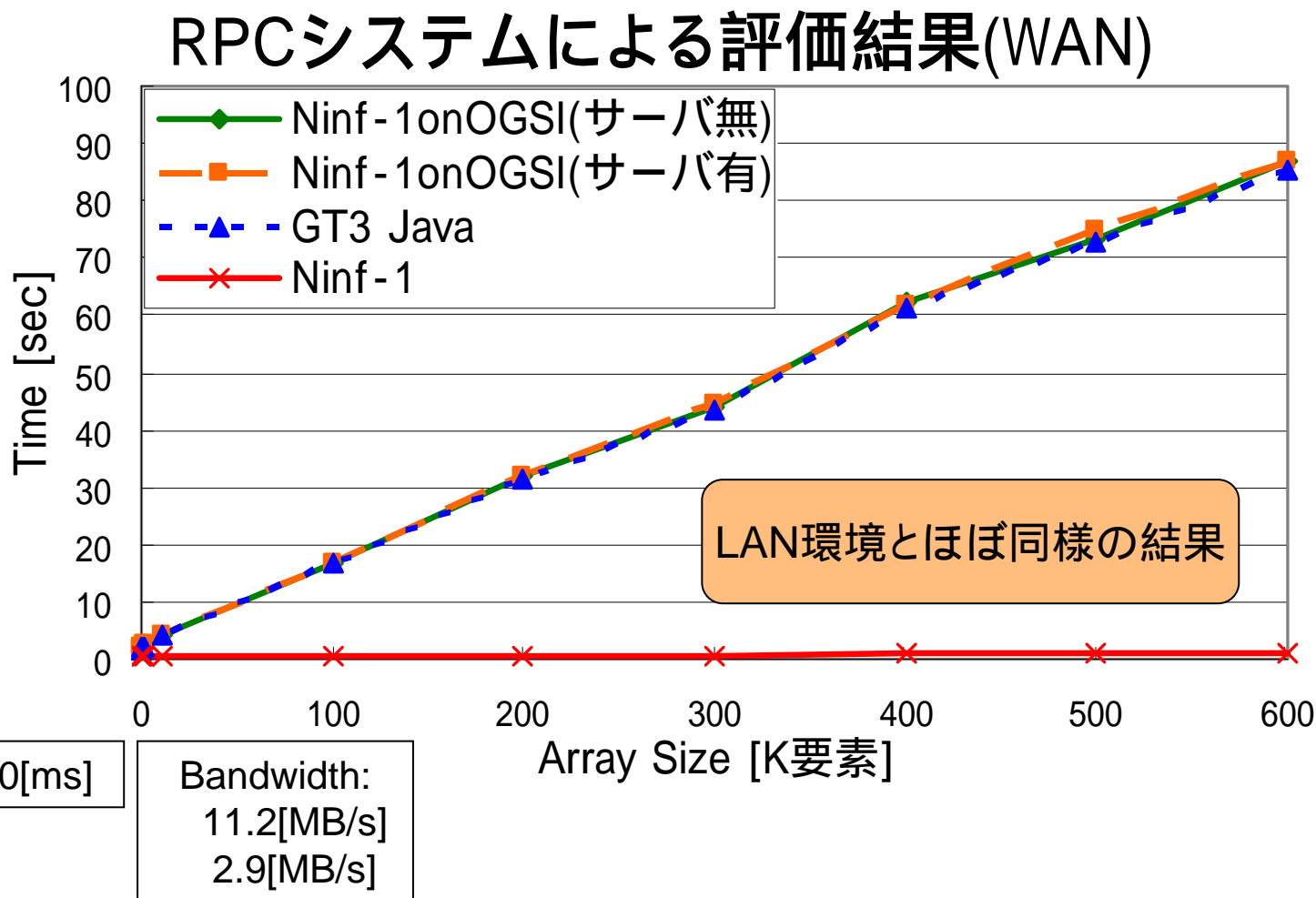
# 比較するRPCシステム



# RPCシステムによる評価結果(LAN)

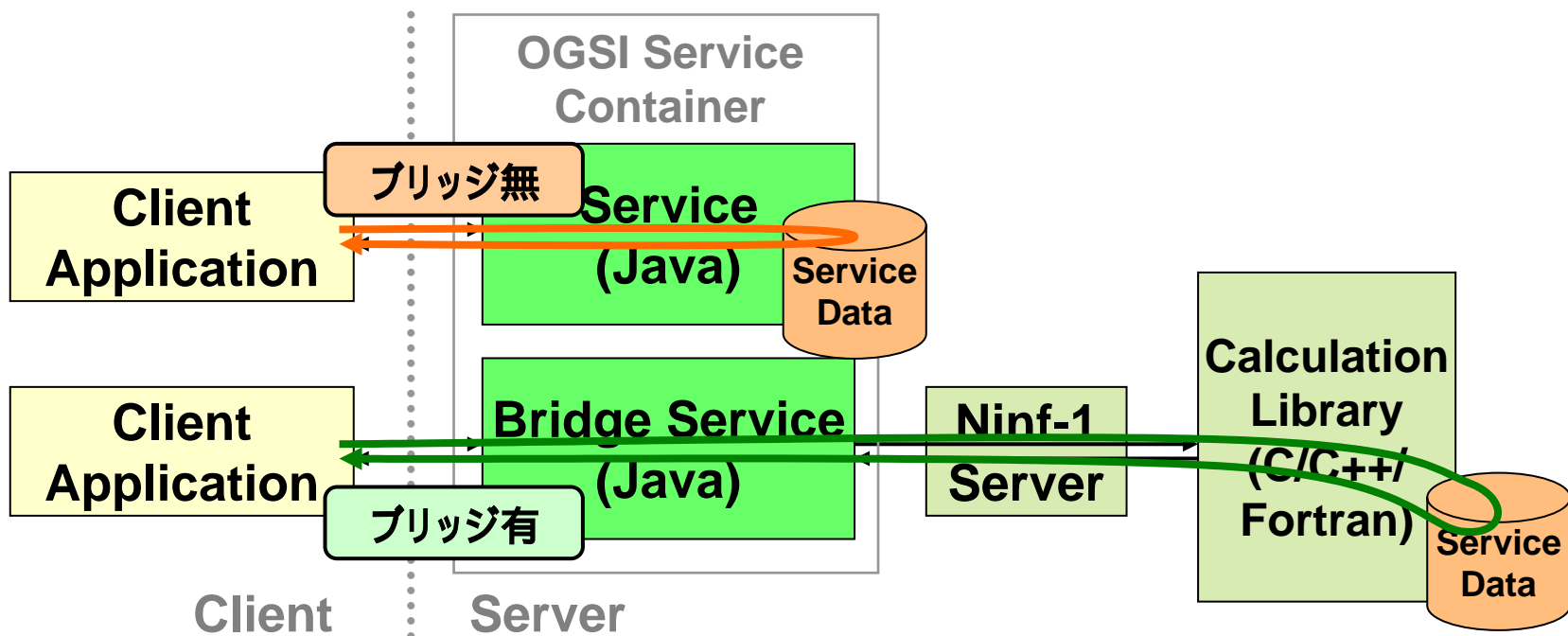


# RPCシステムによる評価結果(WAN)



# サービスデータブリッジの評価

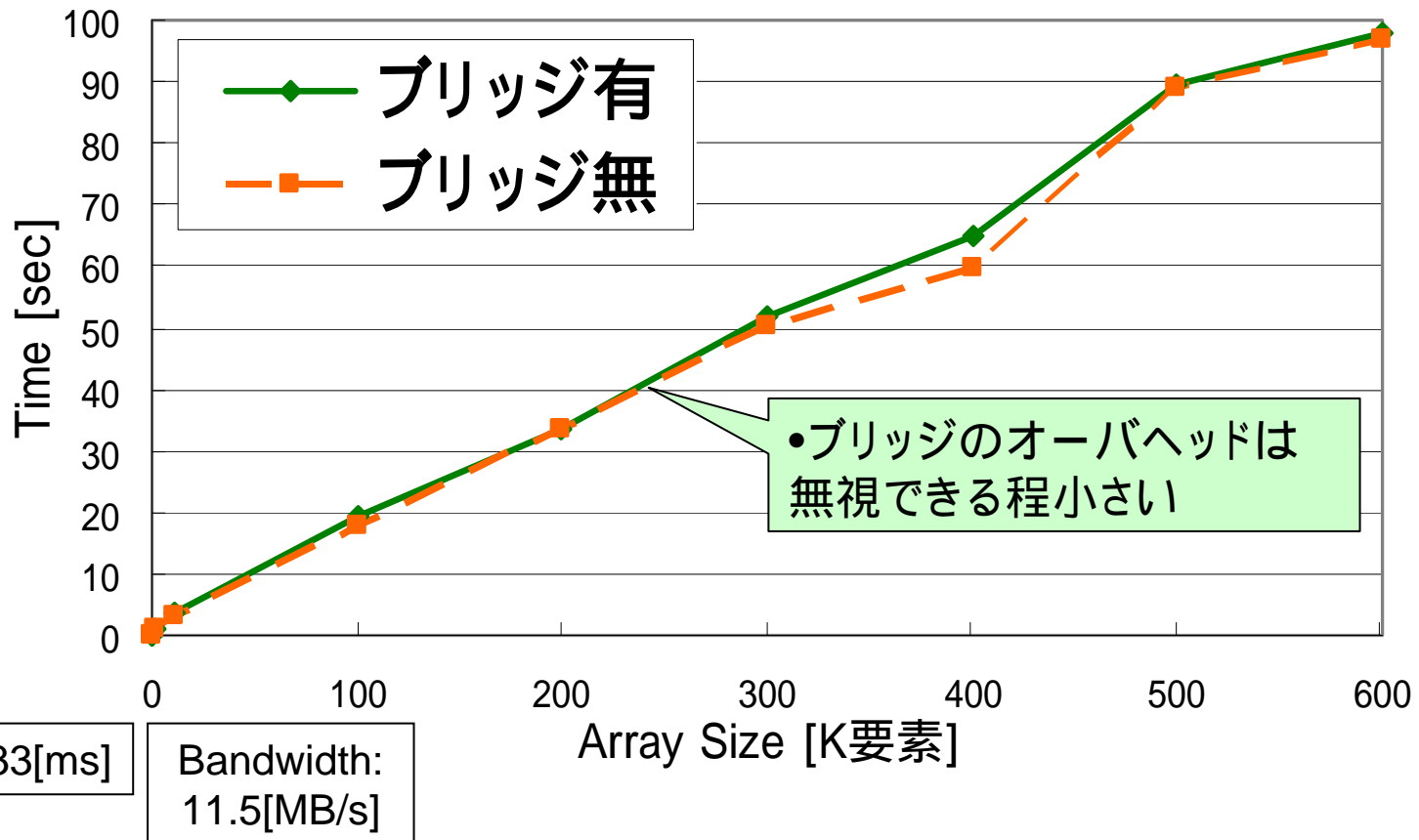
- サービスデータをブリッジする時としない時とで全体のコストを比較評価
  - int型の1次元配列をサービスデータとして送受信



# サービスデータブリッジの評価結果(LAN)



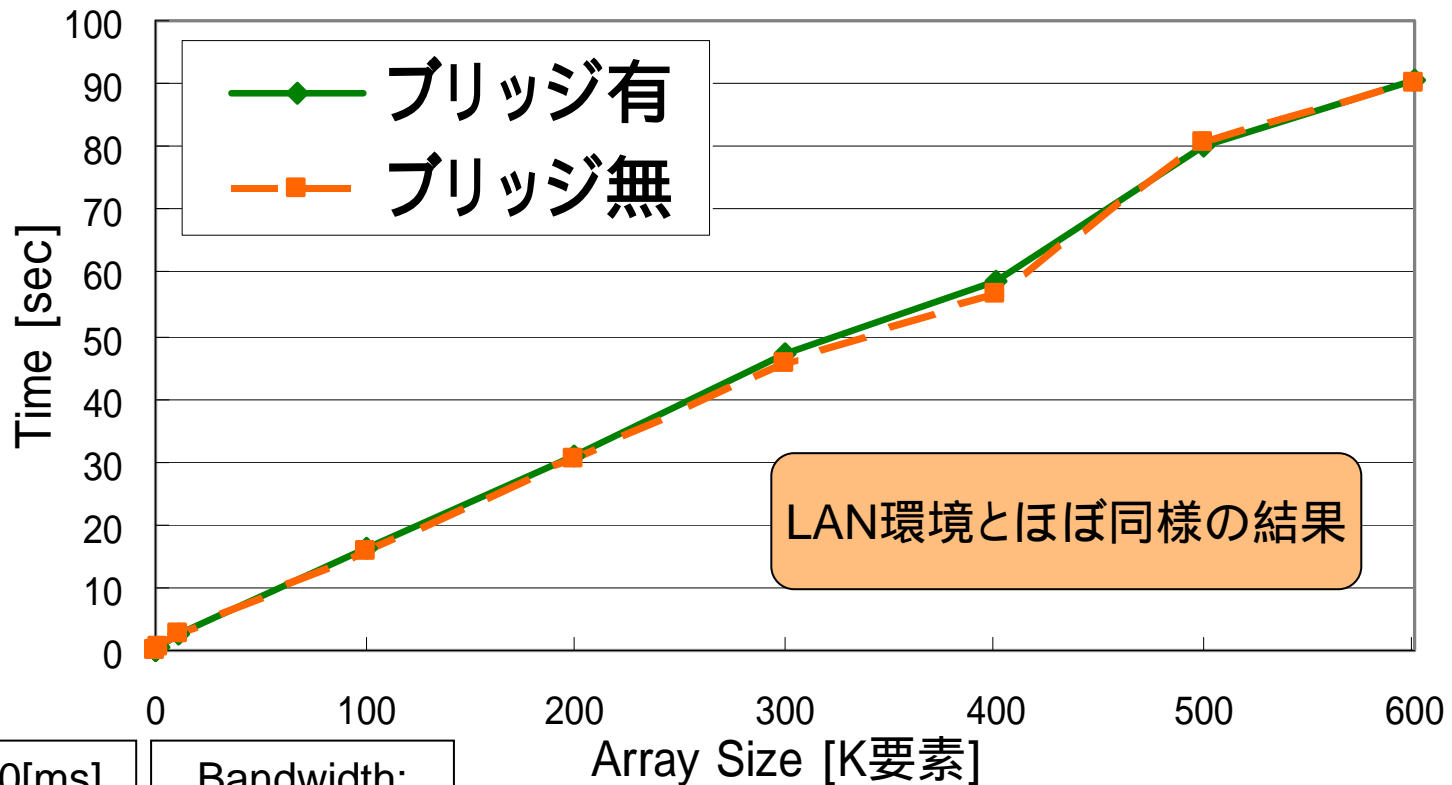
サービスデータブリッジの評価結果(LAN)



# サービスデータブリッジの評価結果(WAN)



## サービスデータブリッジの評価結果(WAN)



RTT:4.0[ms]

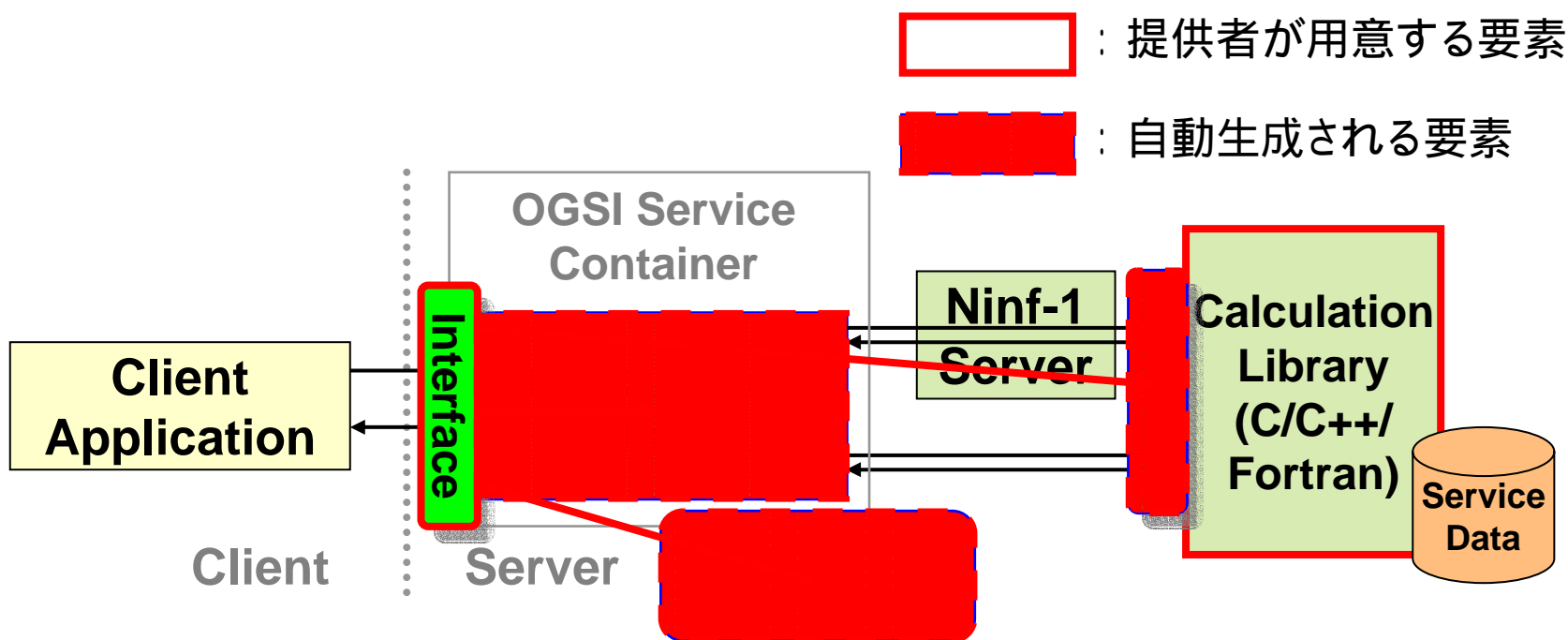
Bandwidth:  
11.2[MB/s]  
2.9[MB/s]



# Ninf-1 サービススケルトンジェネレータの評価

- サービス開発コストの減少

- サービス提供者が用意すべき要素数5 2
- スケルトンは修正せずに利用可能

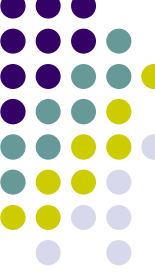


# 考察

- C/C++/Fortran動作環境追加によるオーバーヘッドは小さい
- 開発補助ツールによりサービス開発は十分容易
- GT3の低通信性能からシステム全体としては実用性が低い
  - XMLベースプロトコル通信による性能低下であると考えられる
    - 性能改善手法を提案する研究  
Evaluating Web Services Based Implementations of GridRPC  
[Shirasuna et al., 2002]
    - XML以外の通信プロトコルによる実装

# まとめ

- GT3上にC言語動作環境提供システムを構築
- サービス開発補助ツールの提供
- 構築システムの通信性能評価
  - C/C++/Fortran動作環境追加によるオーバーヘッドは小さい
  - XMLベースプロトコル通信を用いたGT3の通信性能低下によりシステム全体としては実用性が低い



# 今後の課題

- 評価のブレークダウン
  - 通信コスト, XMLパーズングコストなどの詳細化
- XMLベースプロトコル性能改善手法と比較評価
- OGSiの動作環境としての仕様の達成
  - 生存期間管理, 通知など
- 将来提供されるGT3のネイティブなC言語動作環境と比較評価

以降



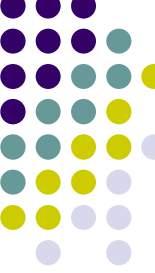
没

# 背景

- 科学技術計算分野での大規模計算の必要性
  - ネットワークインフラの充実
- ↓
- 広域ネットワーク上に分散した計算資源を活用

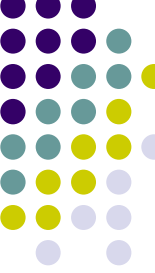
グリッドコンピューティング





# 本研究の目的

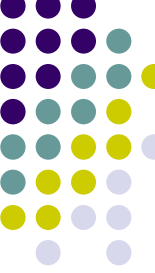
- OGSAにC/C++/Fortranの動作環境を提供
  - OGSIと既存のGridRPCシステムNinfを連携
- サービス開発補助ツールの提供
  - C/C++/Fortranのサービススケルトン自動生成
- OGSAの有用性を検証
  - OGSAベースGridRPCシステムの評価
    - OGSAを実装したGlobus Toolkit 3.0 alpha上でC/C++/Fortran環境のGridRPCシステムを構築
    - OGSAのJava環境でのRPC、Ninfを用いたGridRPCとの比較評価



# Globus Toolkit

- グリッドシステム構築のためのツールキット
  - デファクトスタンダードとして広く普及
- 問題点
  - 独自のプロトコルを採用
    - Globus Toolkit以外では資源の共有が不可能
  - ファイアウォールとの低親和性
    - 双方向のコネクションによりNATを介した運用は困難

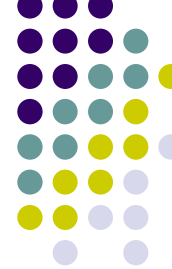




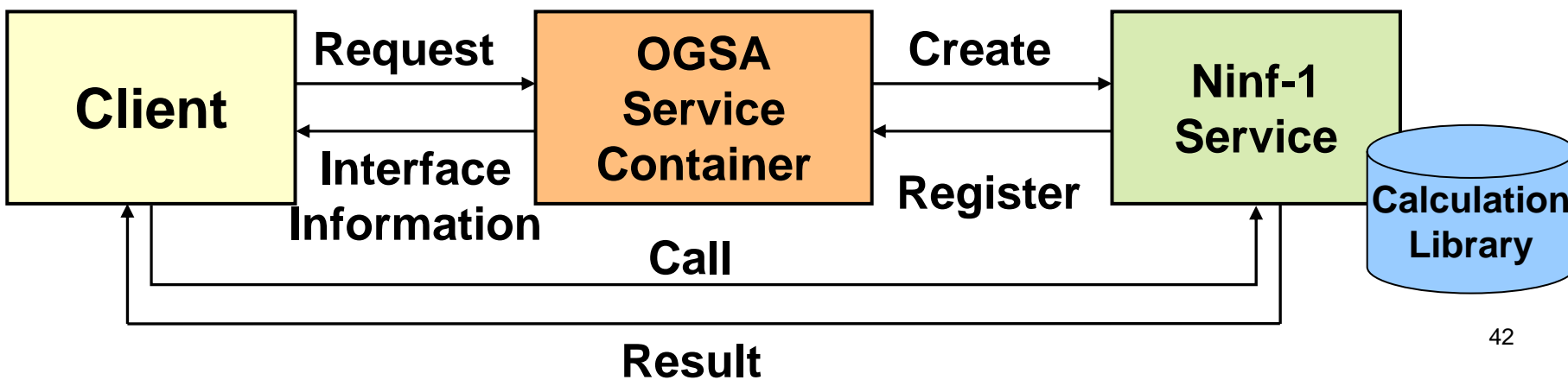
# GridRPC システム

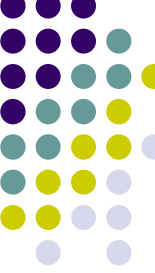
- グリッド上のクライアント・サーバ型RPCシステム
  - Ninf[AIST,TITech]、NetSolve[UTK]
- 科学技術計算分野での利用に適當
  - プログラミングが容易
    - 簡便なAPIを提供
    - RPCの煩雑さを隠蔽
  - 科学技術計算に適したデータ型のサポート
- 問題点
  - 独自のプロトコルを採用
    - 異種のGridRPCシステム間での資源共有が不可能

# 提案するOGSAベースの GridRPCシステムの概要



- クライアント
  - Ninf-1サービス生成要求、サービス呼び出し
- OGSAサービスコンテナ
  - Ninf-1サービスを生成、インタフェース情報通知
- Ninf-1サービス
  - サービス情報の登録、計算結果送信





# Open Grid Services Architecture

- 次世代のグリッドアーキテクチャ
  - GGF4にて発表
  - Globus Toolkit 3.0のアーキテクチャとして採用
- Webサービスのプロトコルを採用
  - XMLベースのプロトコル
    - WSDL サービスのインタフェースを記述
    - SOAP メッセージ交換のエンベロープ

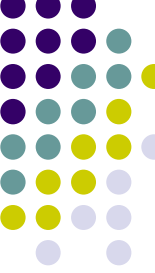
高いインタオペラビリティを確保

# OGSAの問題点(1/2)

- XMLベースのプロトコルによる性能低下
  - バイナリデータからXMLデータへのエンコーディング
    - データサイズの巨大化
    - エンコーディング・デコーディングによるオーバーヘッド



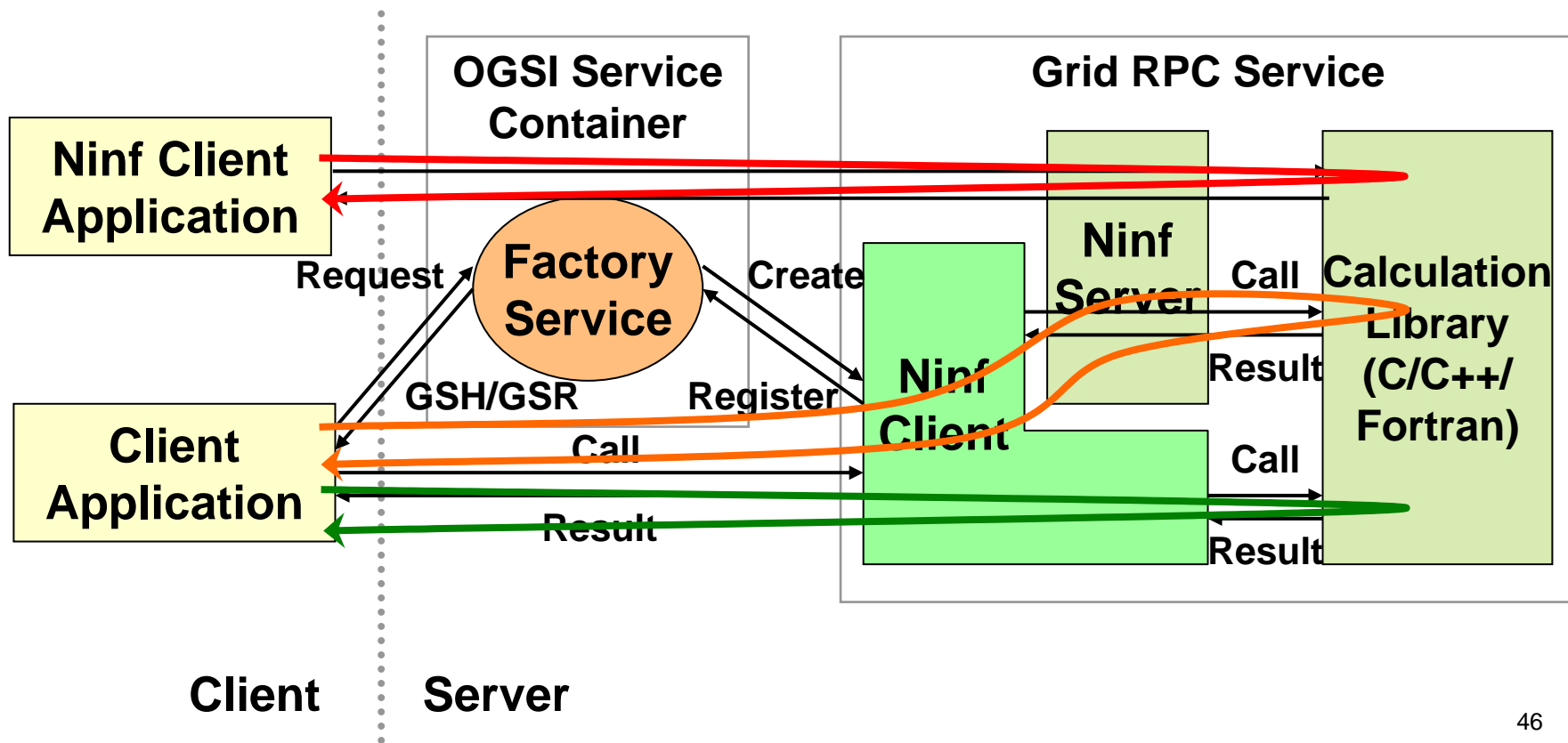
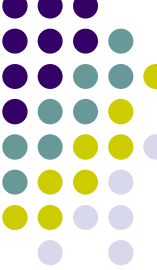
- XMLベースのプロトコルによる性能低下に関する研究
  - Evaluating Web Services Based Implementations of GridRPC [HPDC11 2002, Satoshi Shirasuna et al.]

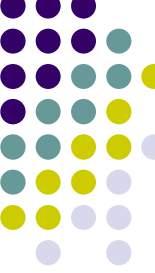


# OGSAの問題点(2/2)

- 動作環境の未整備
  - 現状ではJavaの動作環境のみを提供
    - 他の言語でサービスを提供することが不可能
  - C/C++/Fortranなどの動作環境への高い要望
    - 科学技術計算分野では

# 比較するRPCシステム





# Ninf on OGSiの評価

- 評価環境

- LAN環境

	CPU	メモリ	ネットワーク
サーバ	AthlonXP 1.2GHz	1GBytes	100BASE-TX
クライアント	Athlon 1.3GHz	1GBytes	100BASE-TX

バンド幅	11.5[MBytes/sec]
レイテンシ	0.333[msec]

- WAN環境((a)東工大 (b)産総研)

	CPU	メモリ
サーバ	AthlonXP 1.2GHz	1GBytes
クライアント	PentiumIII 1.4GHz	2GBytes

バンド幅 (a) (b)	11.2[MBytes/sec]
バンド幅 (a) (b)	11.5[Mbytes/sec]
レイテンシ	4.0[msec]

# Ninfサービススケルトンジェネレータの評価



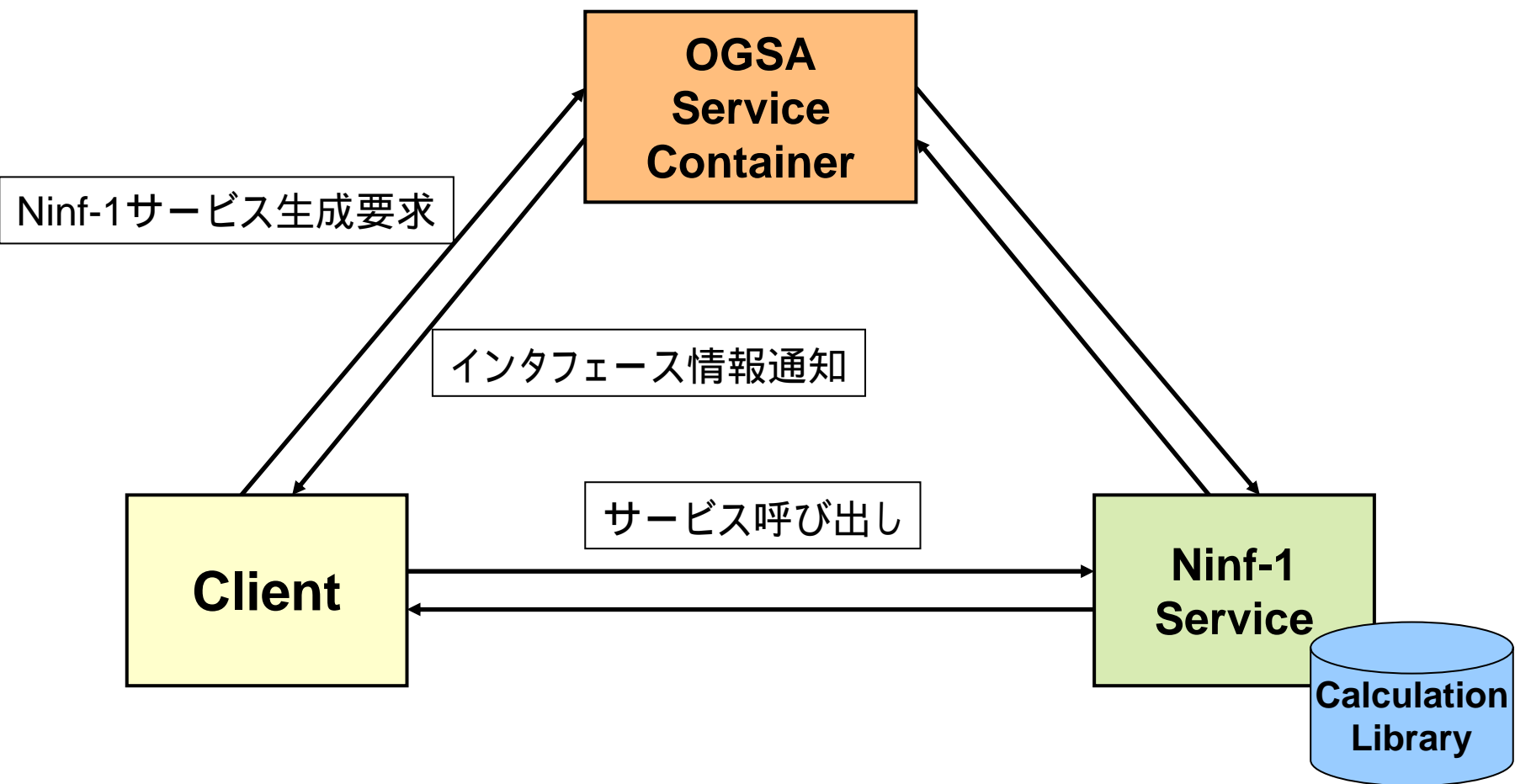
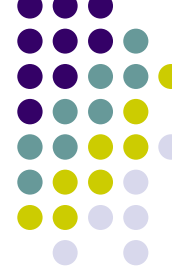
- Ninfサービススケルトンジェネレータの不使用時/使用時にユーザが用意すべきもの

	不使用時	使用時
Factoryサービス		×
Ninfクライアント		×
NinfクライアントのJavaインタフェース		
計算ライブラリ		
計算ライブラリのインタフェース(Ninf IDL)		×
Deployment Descriptor		×
スケルトンの手直し		

サービス開発コストの低下

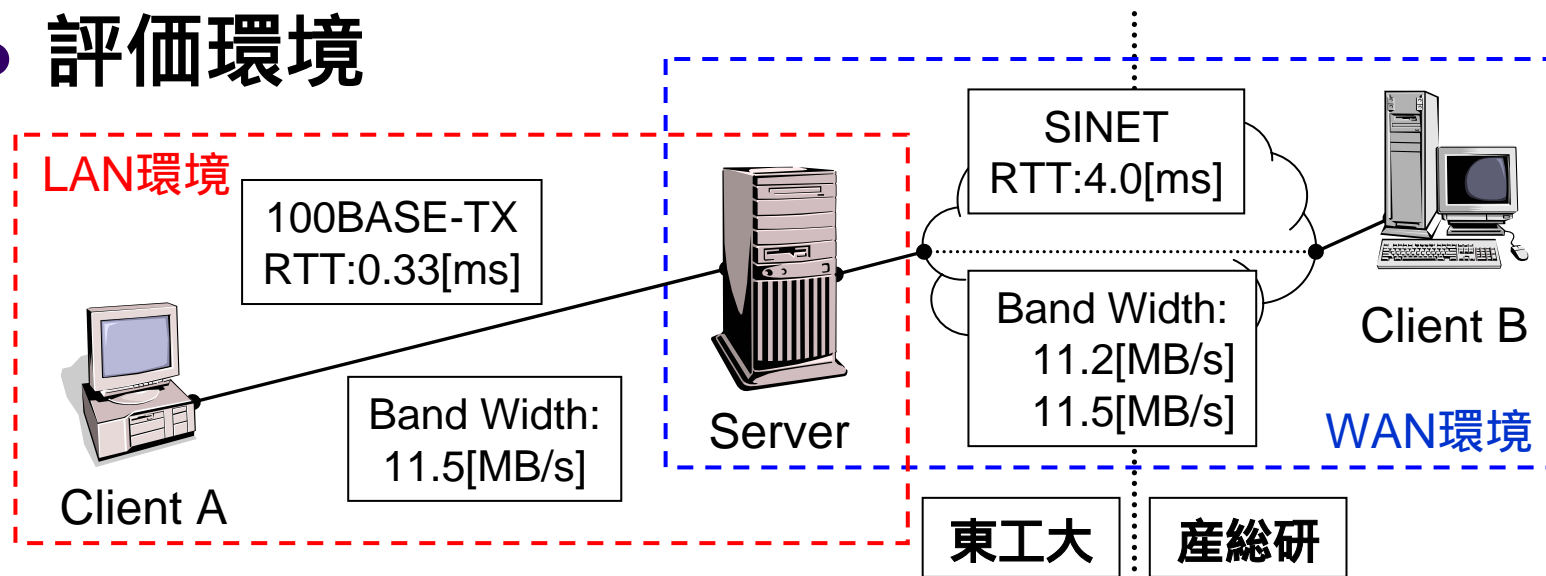


# 提案するOGSAベースの GridRPCシステムの概要



# GridRPC on OGSAの評価

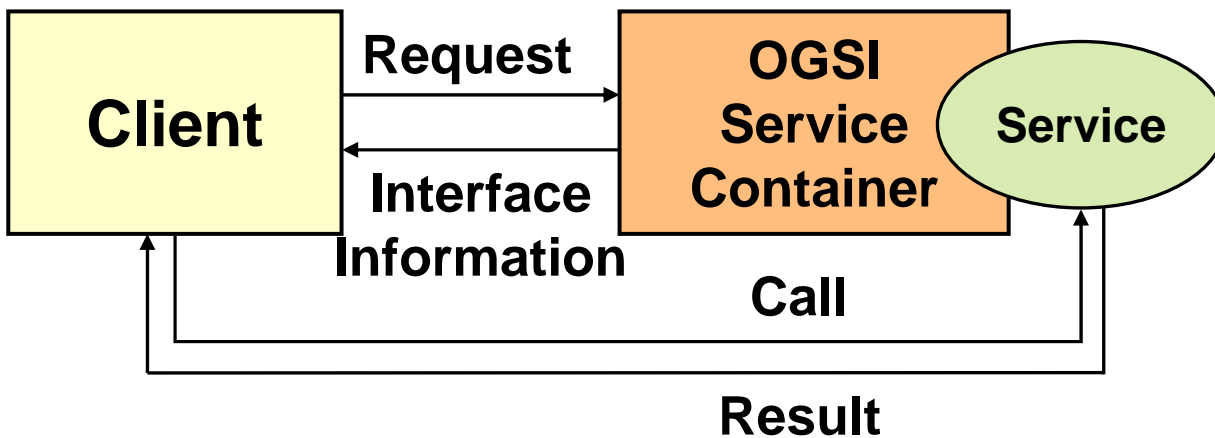
## ● 評価環境



	CPU	メモリ
サーバ	AthlonXP 1.2GHz	1GBytes
クライアントA	Athlon 1.3GHz	1GBytes
クライアントB	PentiumIII 1.4GHz	2GBytes

# OGSAの概要

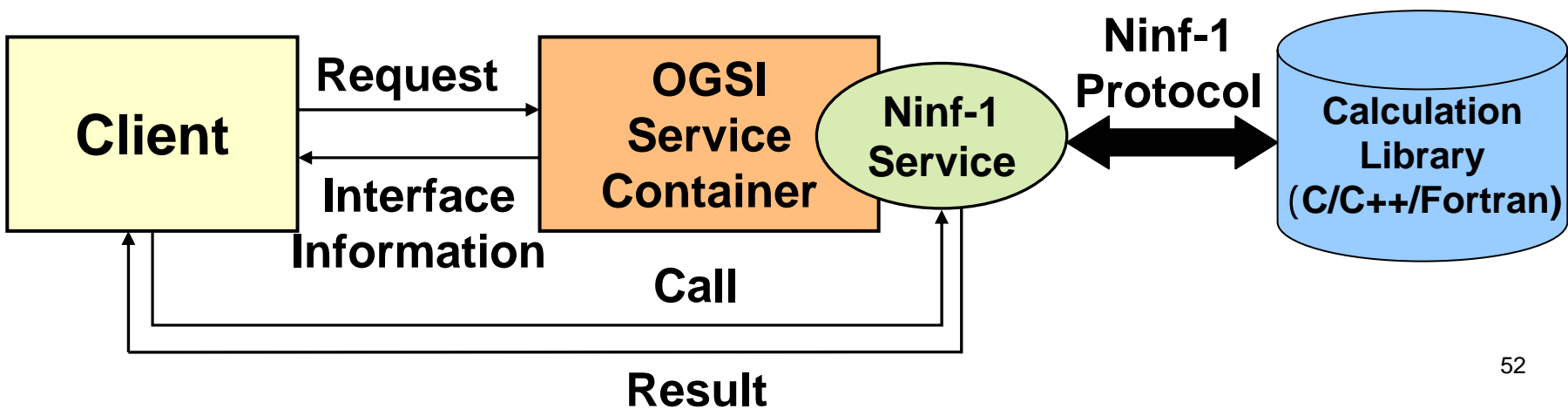
- OGSAサービスコンテナからサービスを起動
  - クライアントとサービスが通信
- 現在OGSAサービスコンテナはJavaのみ対応
  - サービスはJavaでのみ記述可能



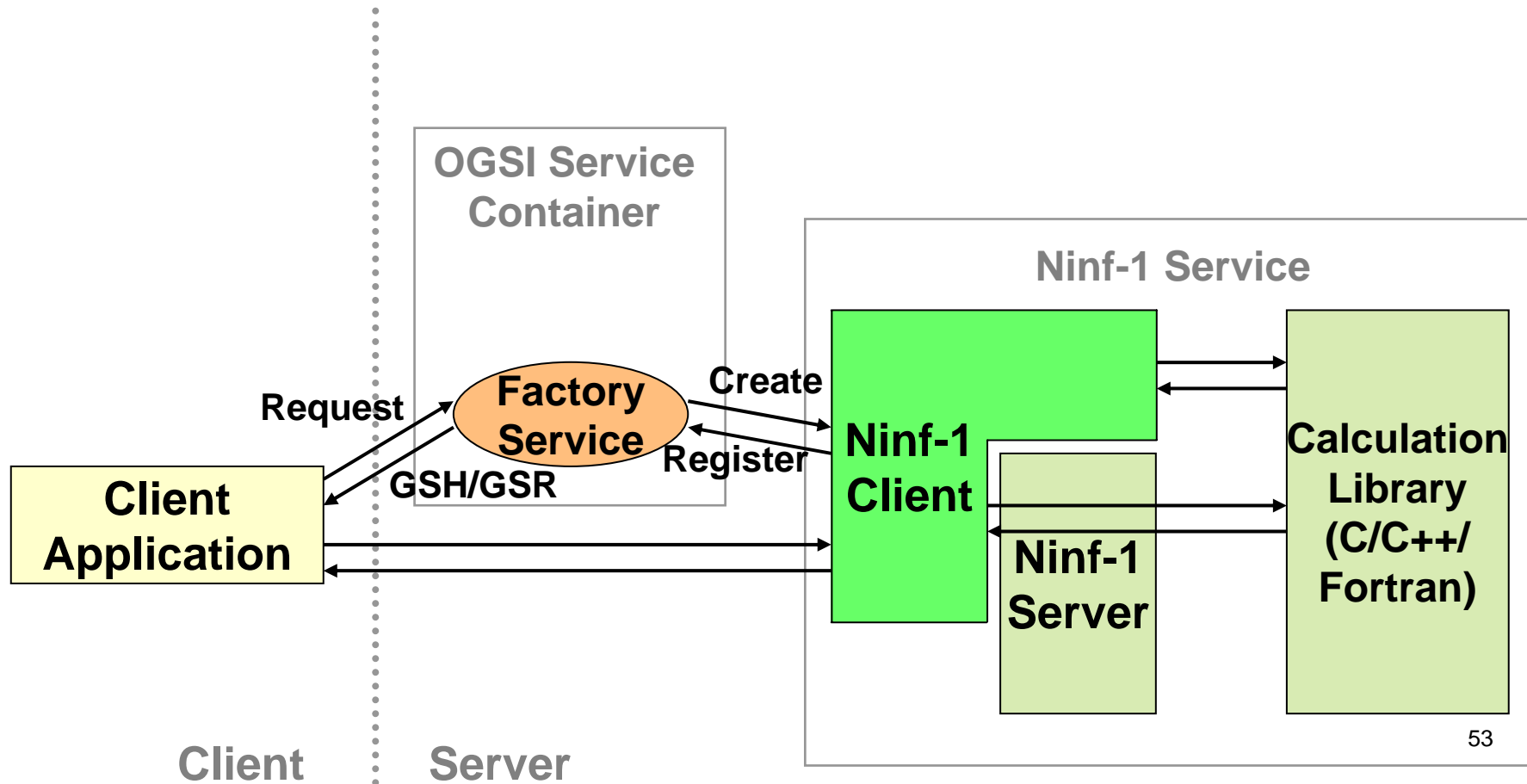
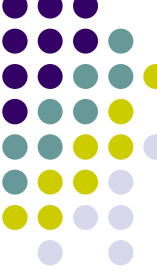
# 提案するOGSAベースのGridRPC システムの概要



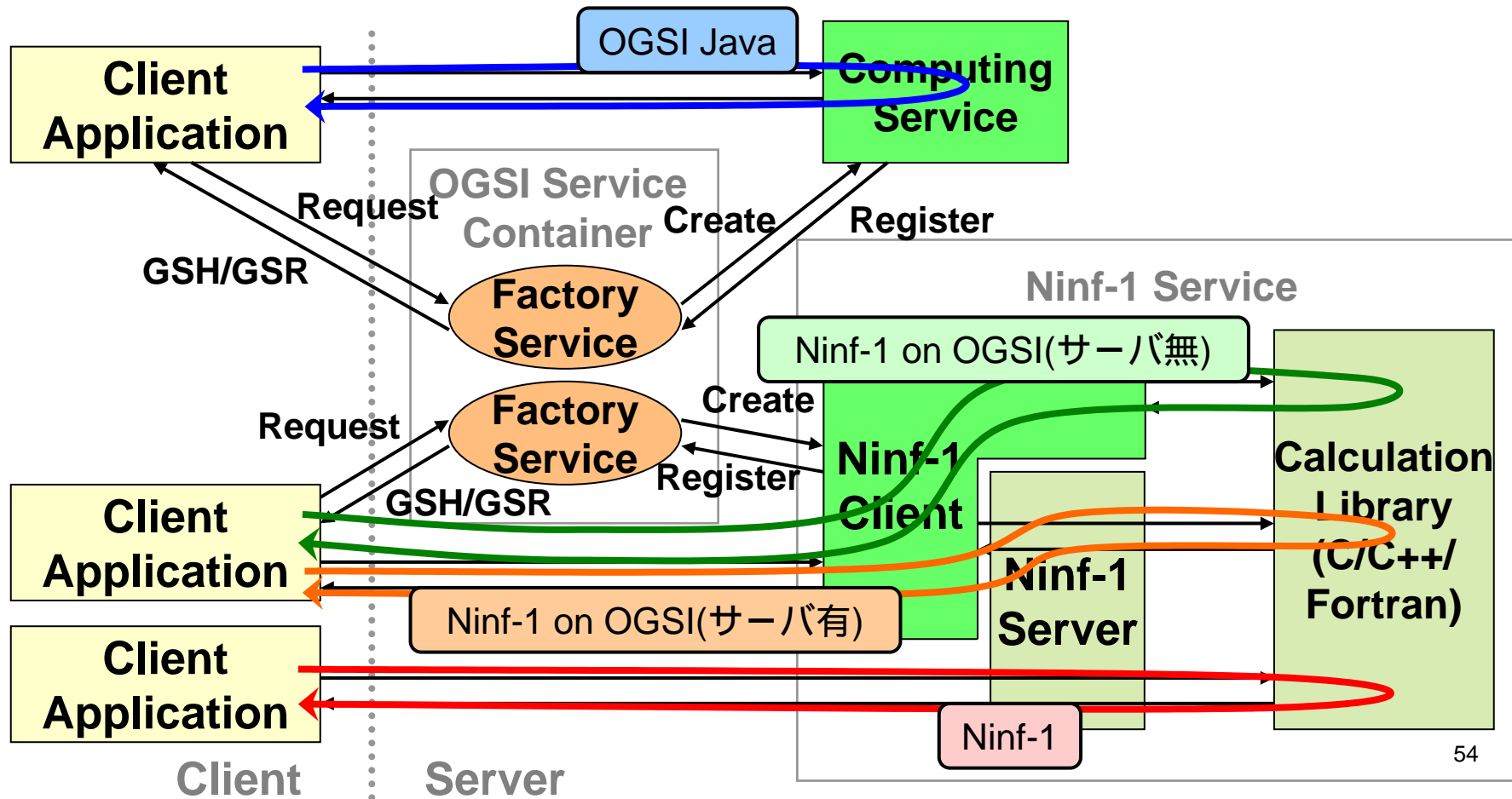
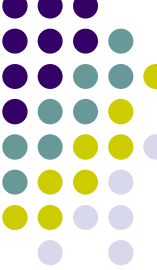
- GridRPCシステムNinf-1をローカルに用いてJavaからC/C++/Fortranへの呼び出しを実現
- クライアントはNinf-1サービスと通信
  - Ninf-1サービスはNinf-1クライアントとして計算ライブラリと通信



# GridRPC on OGSA



# 比較するGridRPCシステム



# サービス提供者が用意する要素

