

オープンソースカンファレンス2008 Nagoya

Struts, Springを活用した 業務システム構築事例と 成功のポイント

2008年8月9日

野村総合研究所 情報技術本部
オープンソースソリューションセンター

松野 洋希

株式会社 野村総合研究所 情報技術本部 オープンソースソリューションセンター(OSSC)

Mail : osscc@nri.co.jp Web: <http://openstandia.jp/>

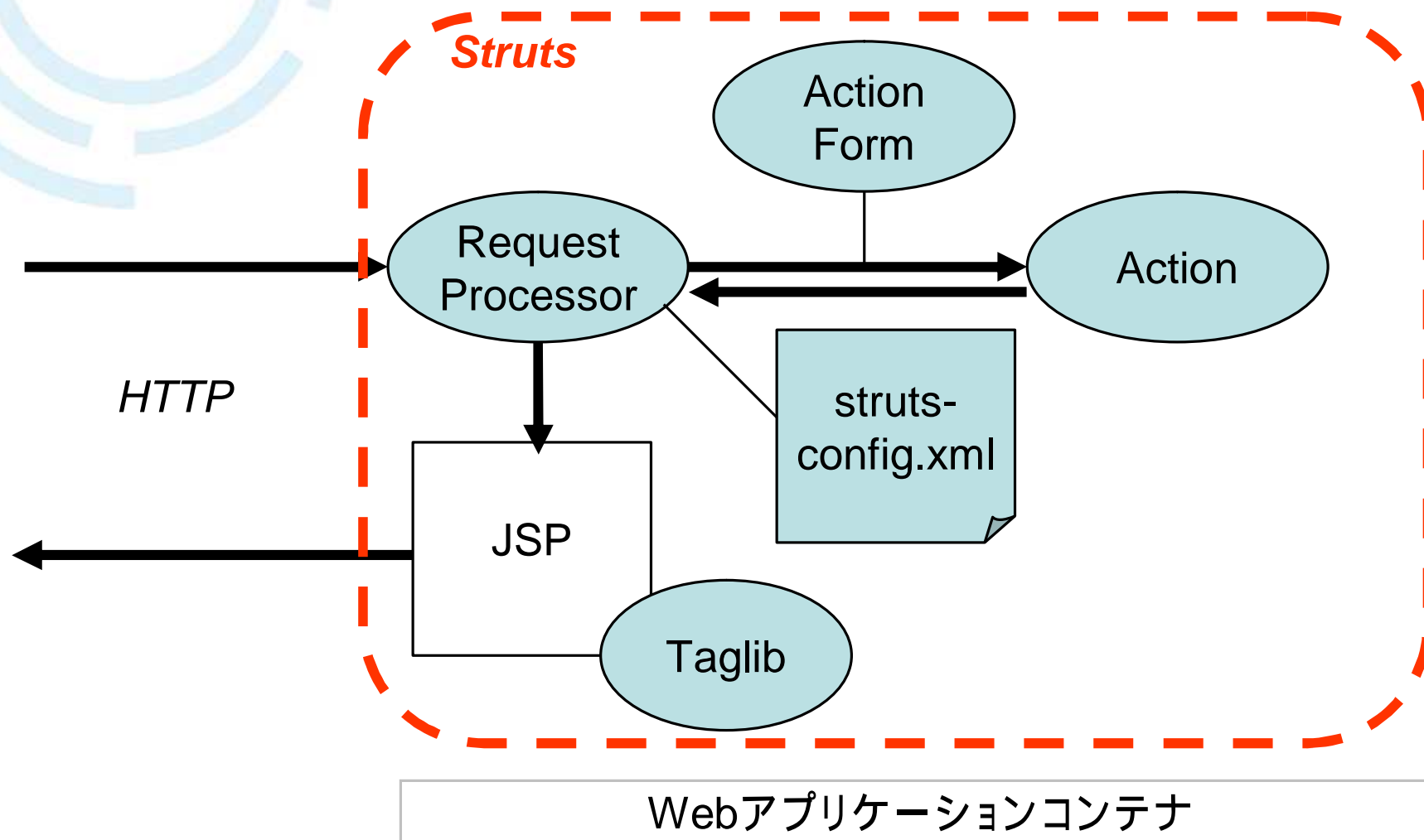


自己紹介と会場アンケート

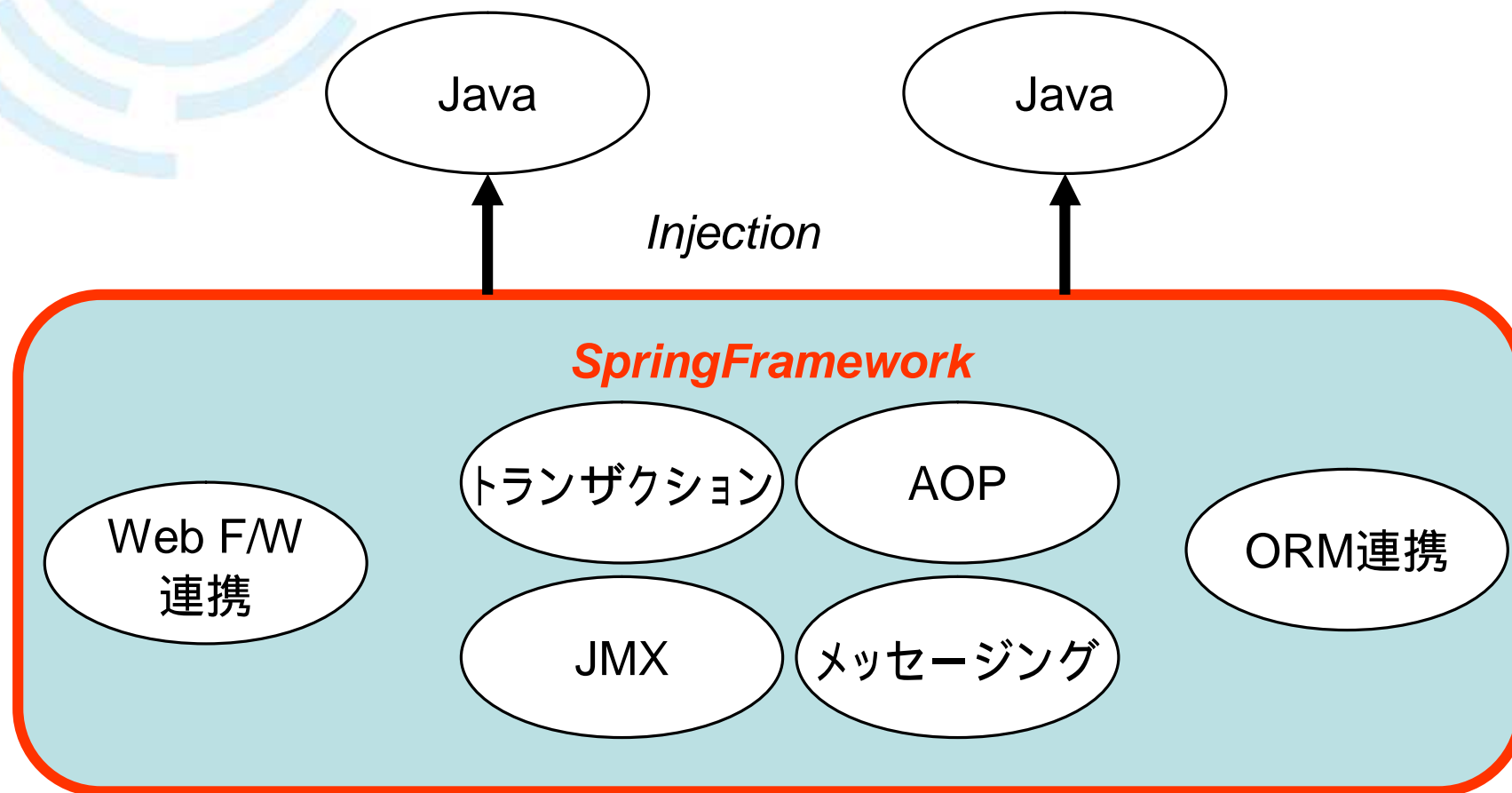
(事例の前に) オープンソースの動向

オープンソースソフトウェアの おさらい

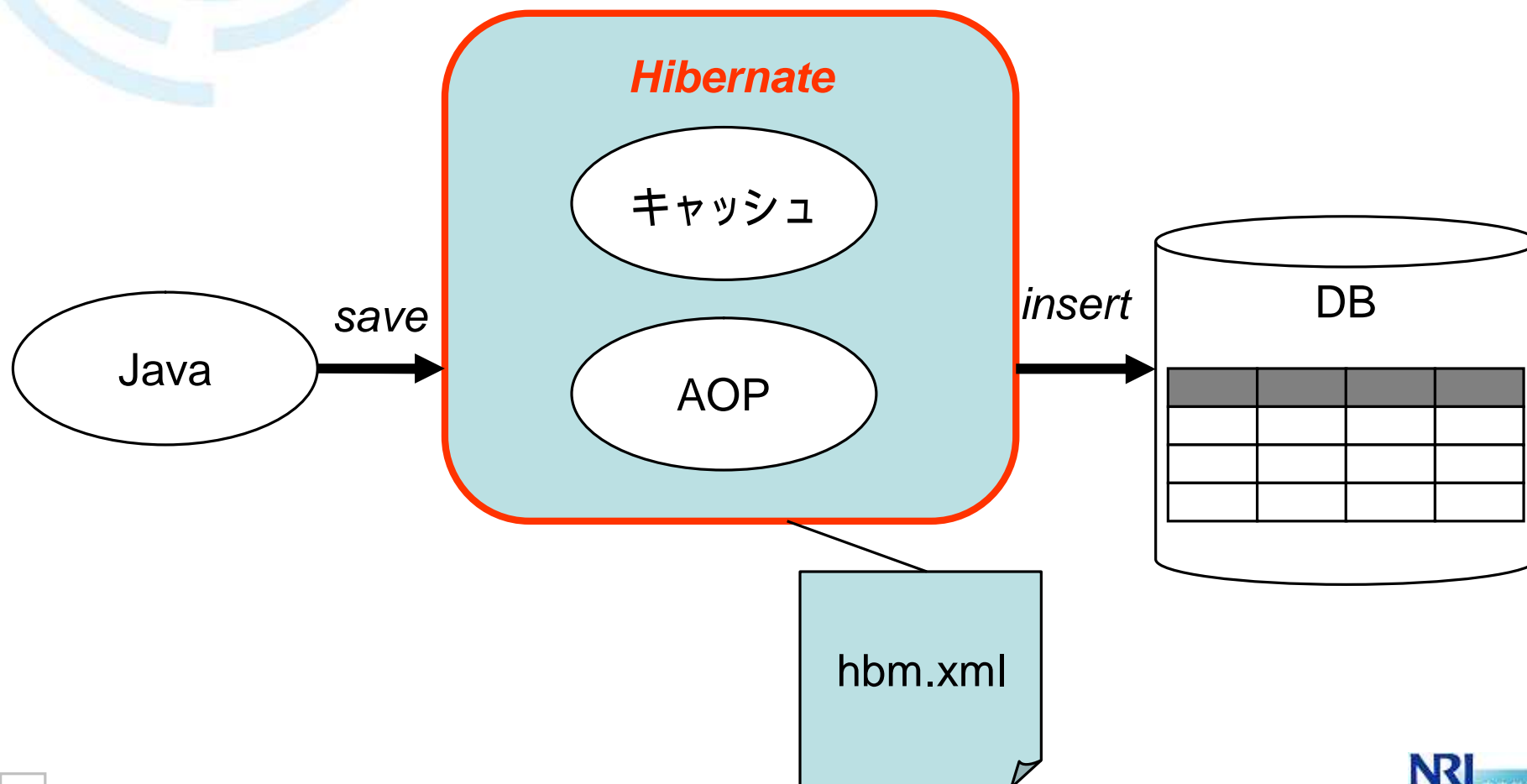
MVC型Webアプリケーションフレームワーク



● 軽量DIコンテナ



●O/Rマッピングフレームワーク



導入事例1： 不動産情報提供サイト

● プロジェクトの概要

- 不動産情報を提供するインターネットサイト
- 50万ページビュー / 日

● 開発体制

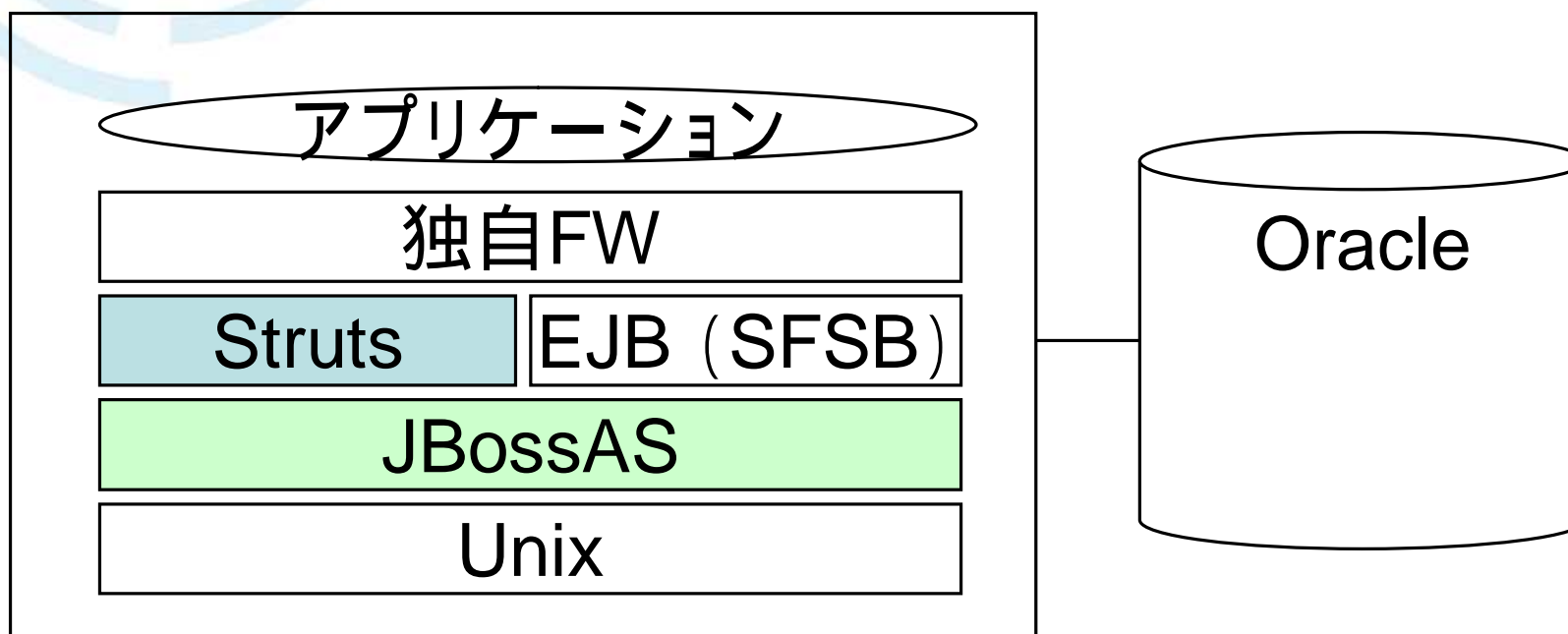
- ピーク時は100人体制での開発

● オープンソースの採用

- 初期コスト 数千万円 0円
- 保守コスト 数百万円 / 年 約半額

WebAPサーバ

DBサーバ



標準化のポイント …その前に

● 目的

- ユーザから見た操作感の統一
- 品質の均質化

● 対象範囲

- 画面デザイン（フォーマット、色、文言、・・・）
- アプリケーションの構成
- 開発方法（開発環境、プログラム、・・・）
- ネーミングルール、コーディングルール
- 共通部品

アプリケーションチーム

アプリケーションの開発
APL共通部品
設計書標準化

アプリケーション基盤チーム

開発プロセス標準化
アプリケーション方式
(認証、文字コード、
メッセージング…)

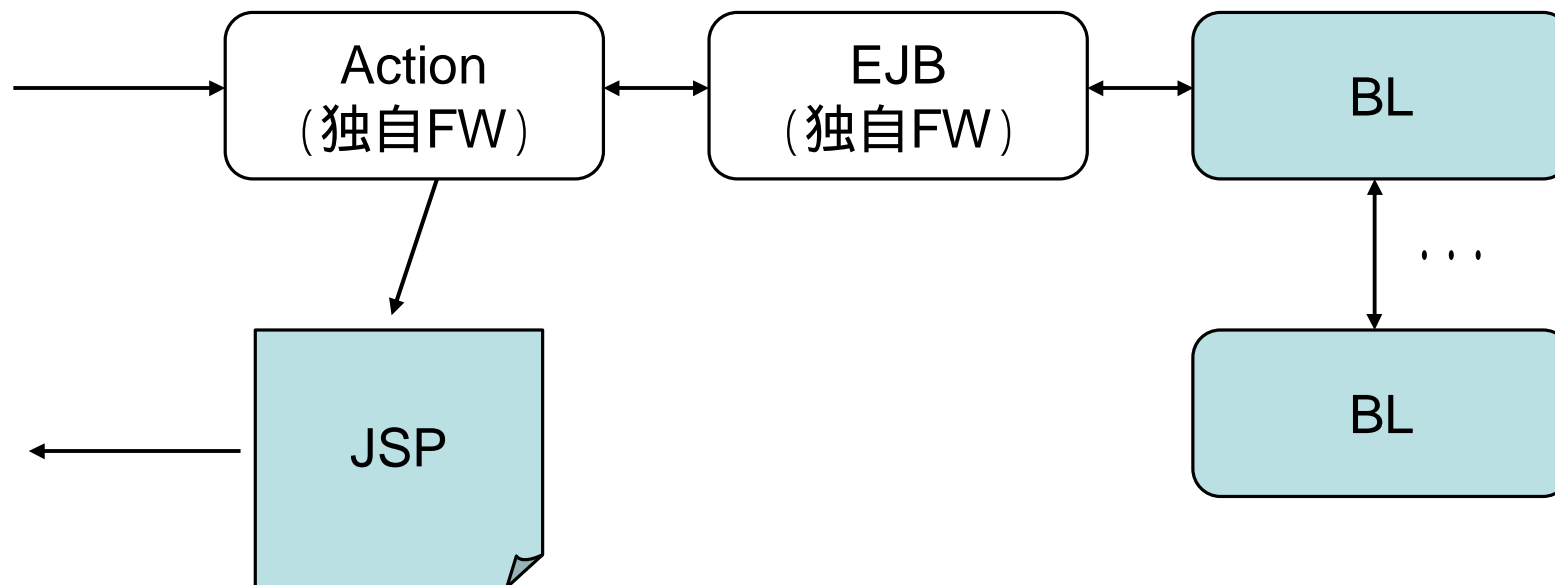
基盤チーム

基盤設計・構築
環境設計
運用設計
性能テスト、障害テスト

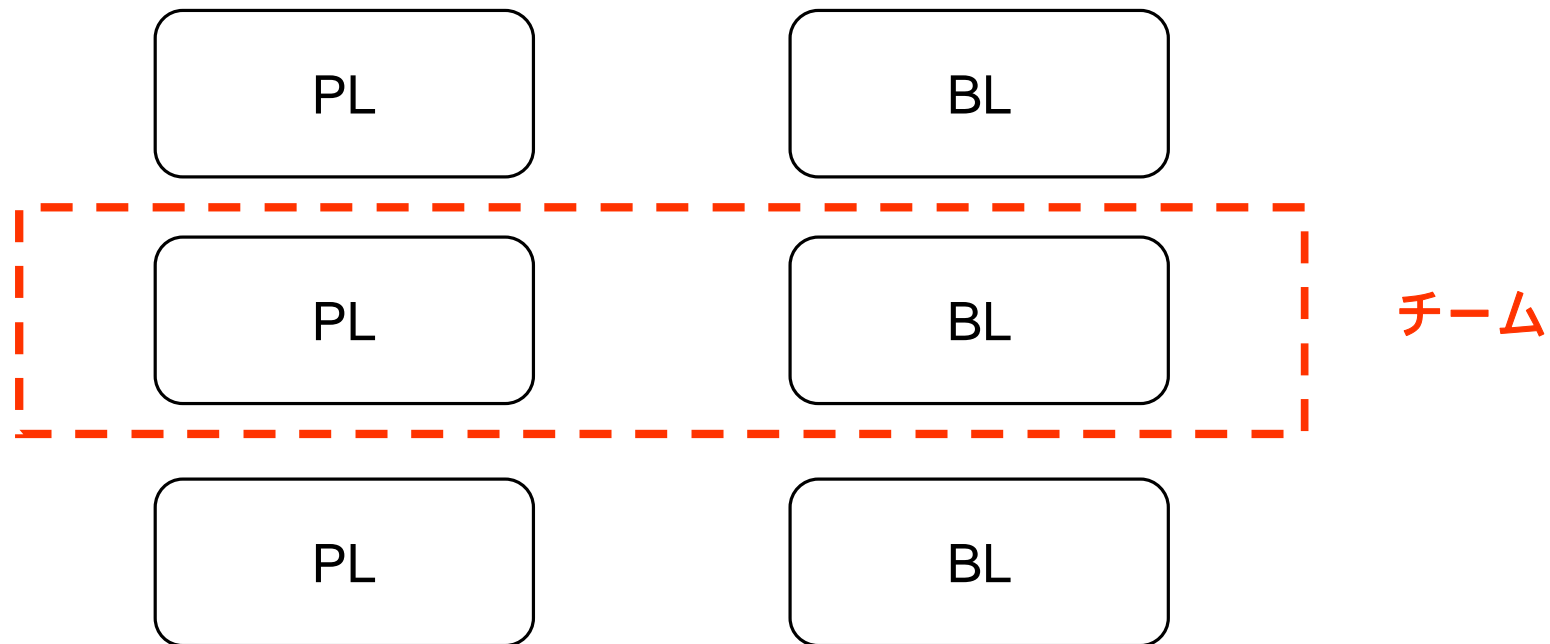
●問題点

- COBOLに慣れた開発者が多く、Java開発に慣れていない

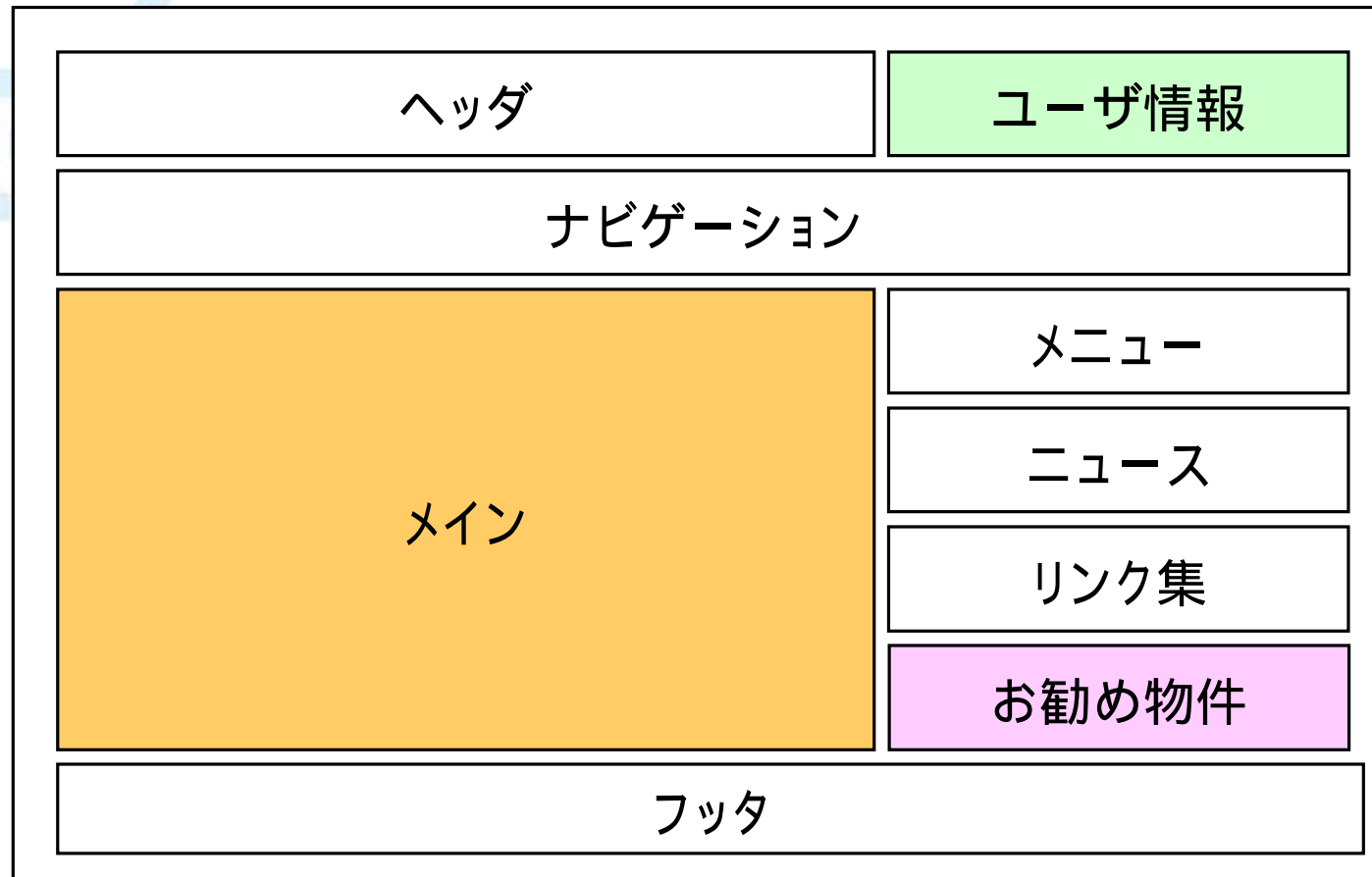
- トランザクションスクリプトモデル
 - 手続き型処理
- 独自FWによるEJB呼び出し処理の隠蔽



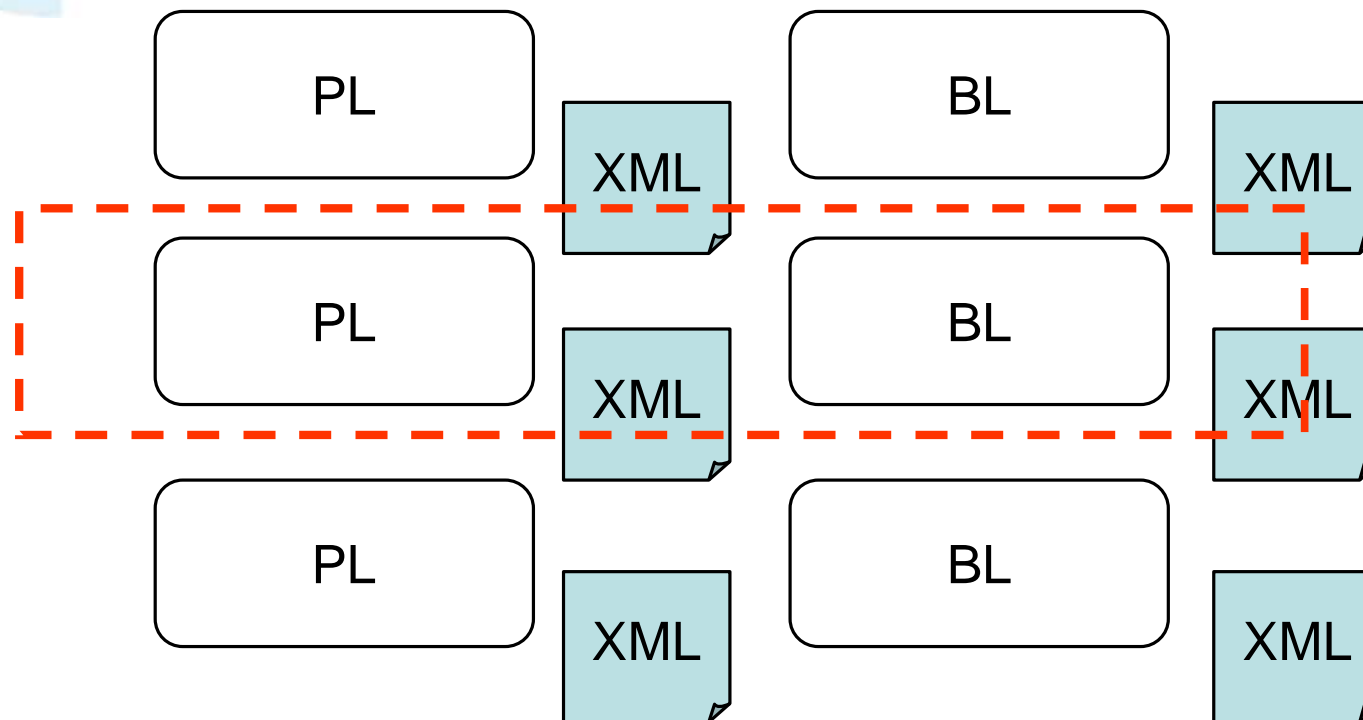
- アプリケーションチームからの要望
 - 画面のインタフェースを統一したい
 - 機能で分割したチーム分けにしたい



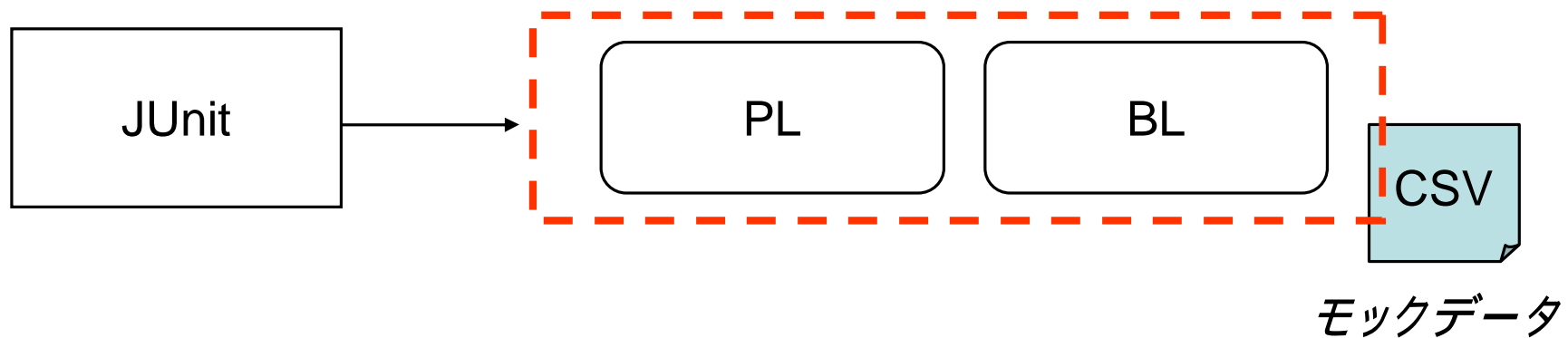
● Strutsのタイルシステムを利用



- 定義ファイルを分割
- ソース管理で競合を避ける



- PL + BLを一気にテスト
 - 画面単体テスト、Java単体テストを省略
 - 工数削減
- モックを利用することで開発端末でも実施可能



●情報共有に重点

- メーリングリストは全員参加
 - 話題で宛先を変える（開発者、チーム別、・・・）
 - コミットメール
 - 課題システムメール（登録、割当、差戻、・・・）
- Wiki導入
 - 開発ガイドライン
 - ページビューをあげる仕組み（俗語辞典、人物列伝、・・・）

導入事例2： ASPオンライントレードサイト

● プロジェクトの概要

- 証券会社のオンライントレードサイトをASPサービスとして提供
- 最大 100ページビュー / 秒

● 開発体制

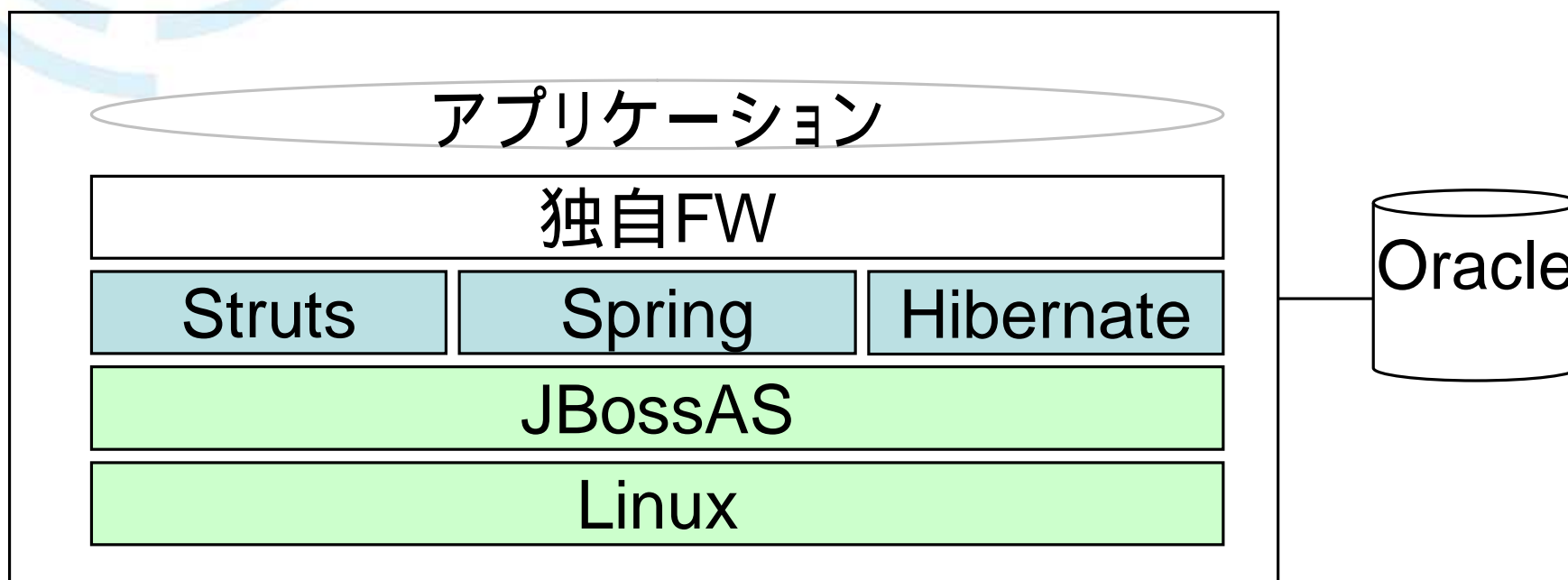
- 日本 + 中国 3拠点 分散開発

● オープンソースの採用

- 自社ASPなのでコストを下げたい
- サーバの増減を気軽に

WebAPサーバ

DBサーバ



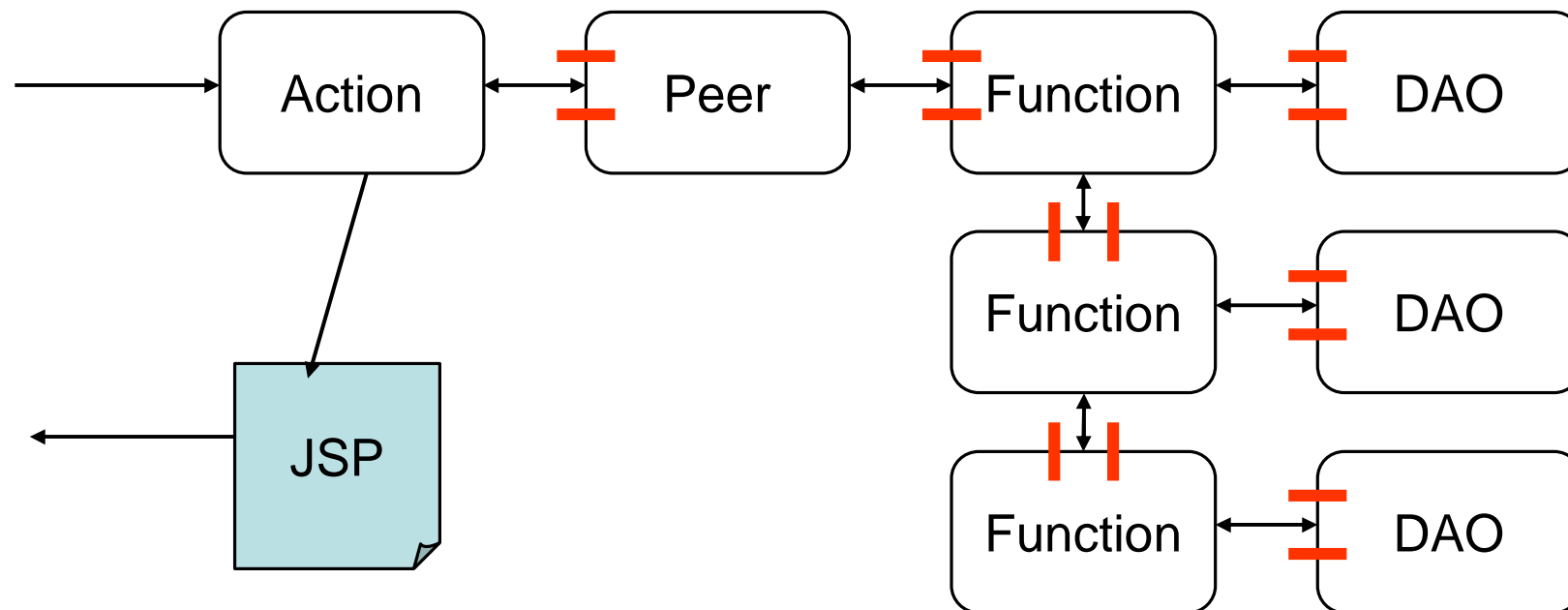
●問題点

- 日本 + 中国3拠点の分散開発



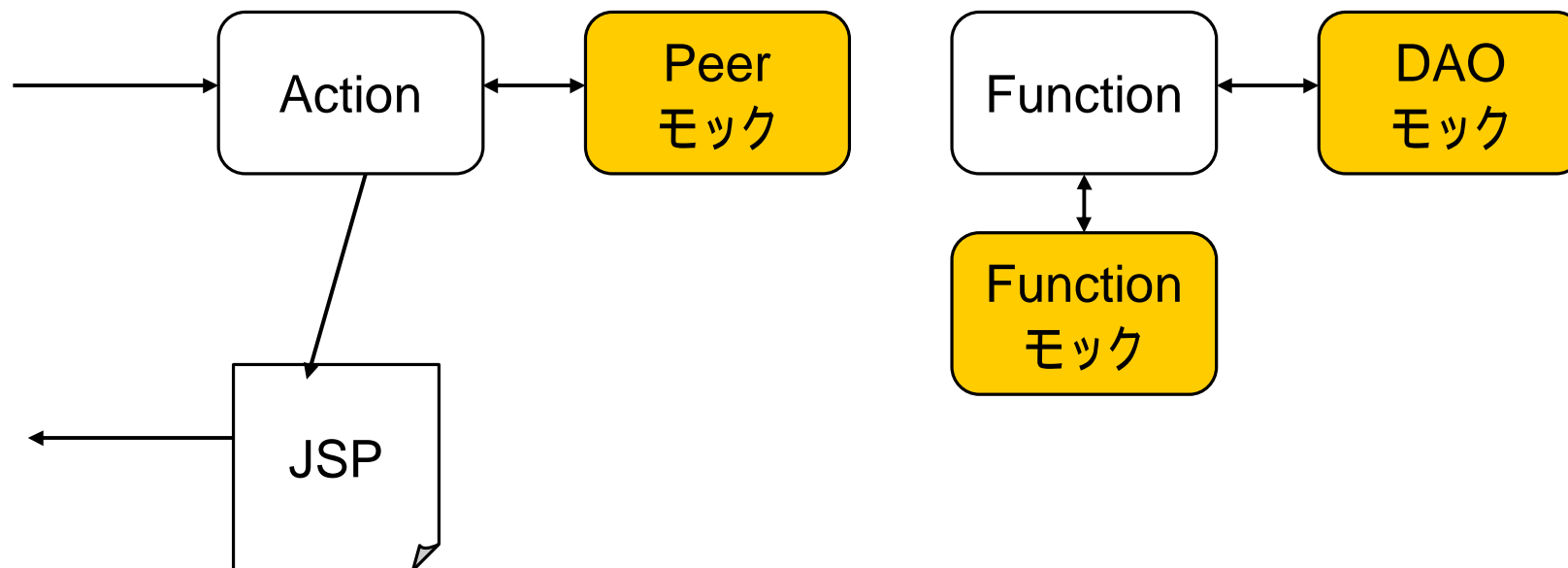
● インタフェース指向

- インタフェース配布により結合時の不整合発生を低減
- SpringFrameworkの利点



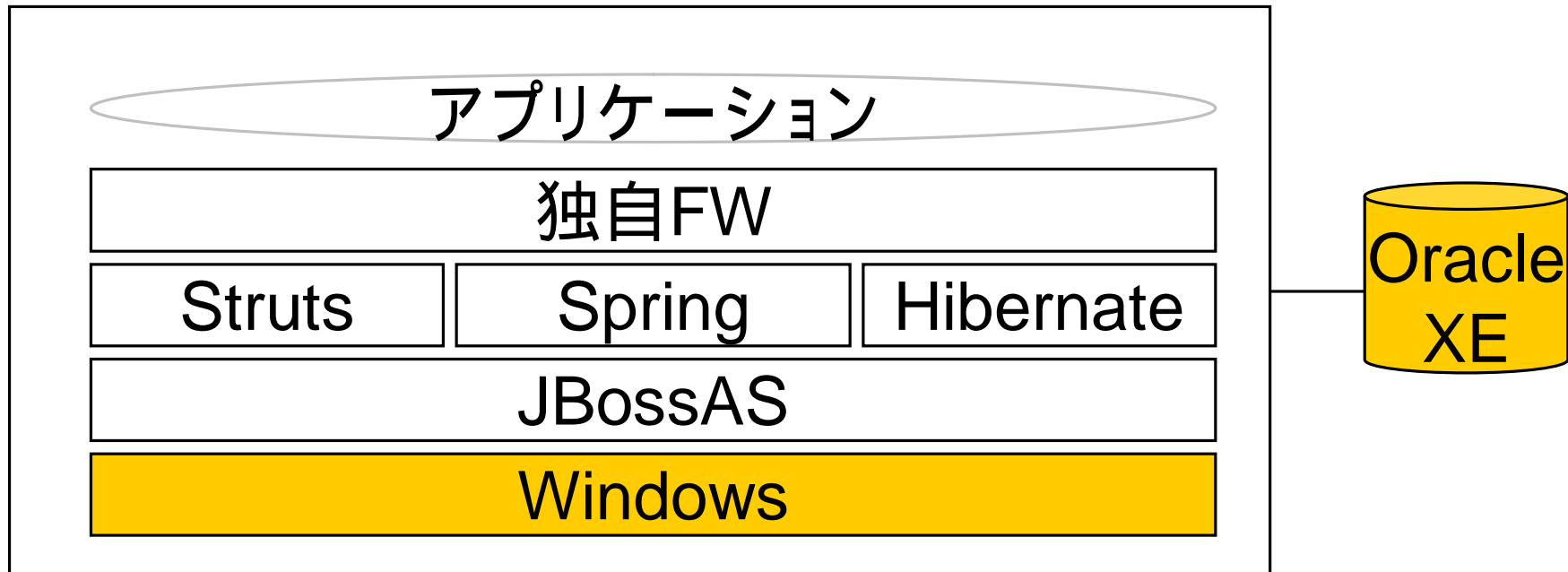
●単体テスト

- モックを利用することで拠点内でテストを実施
- SpringFrameworkの利点



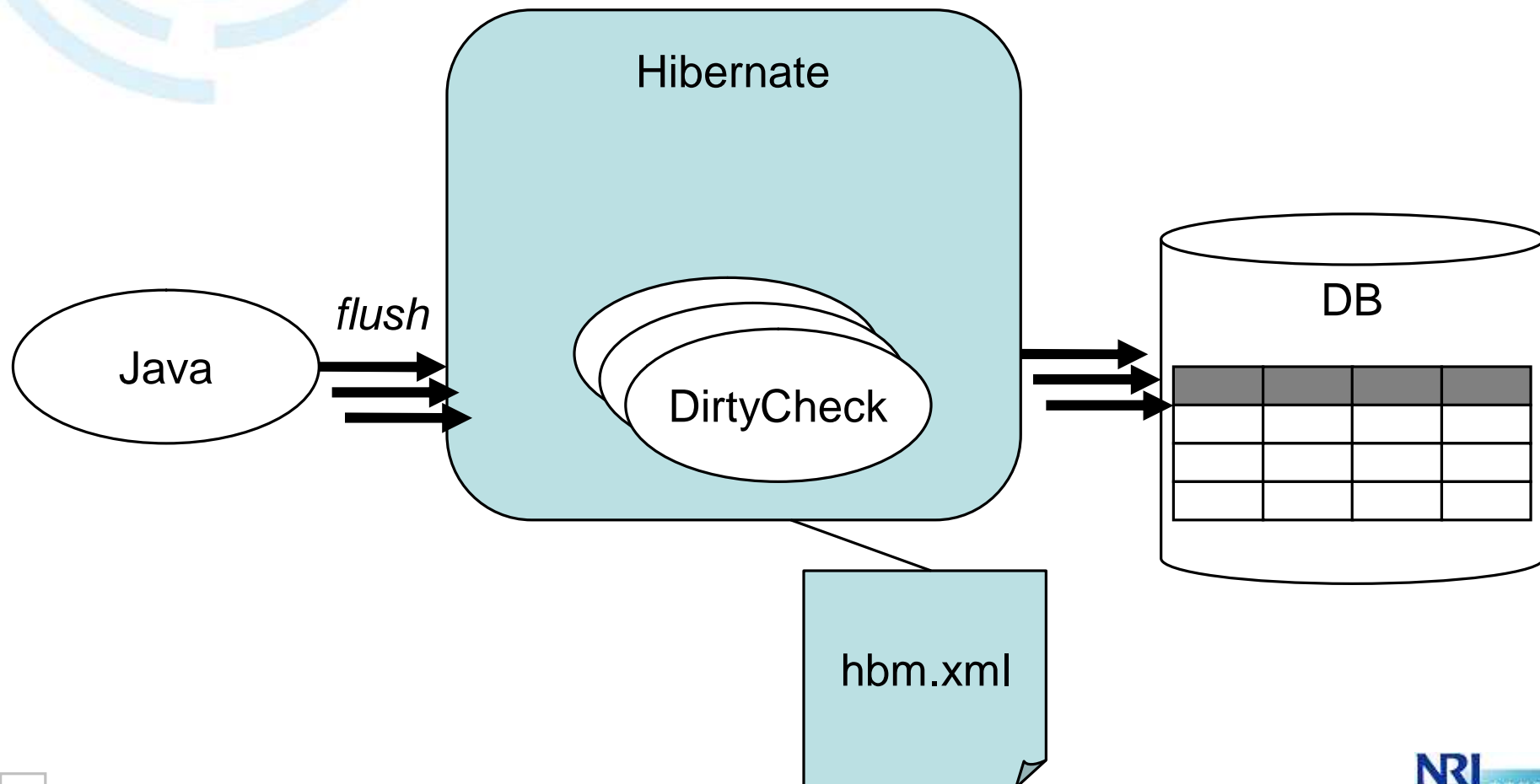
● 連結テスト

- 拠点内で疎通レベルの連結テストを実施
- Javaの利点 (run anywhare)
- OracleXE の利用

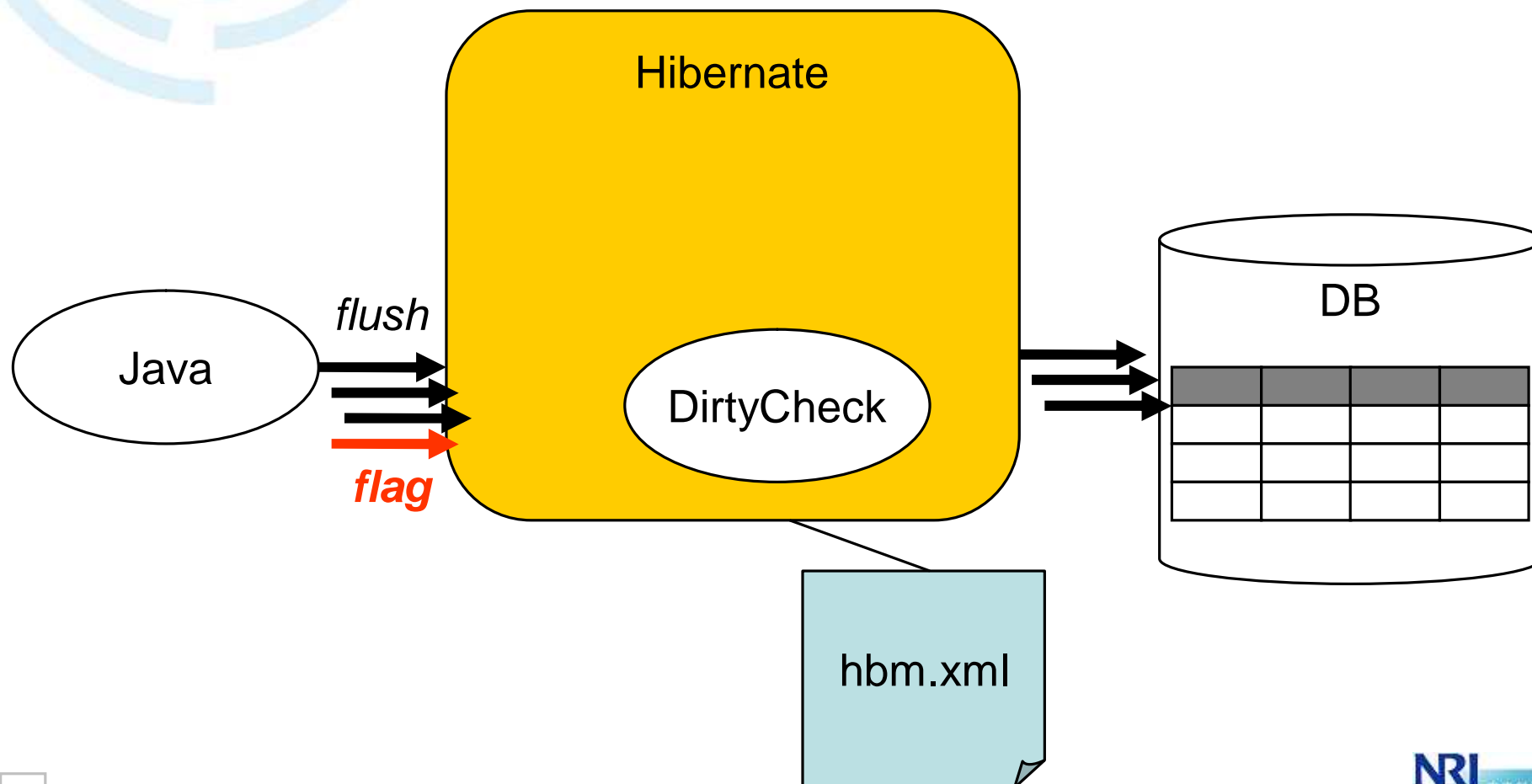


● 基盤チームからの問い合わせ

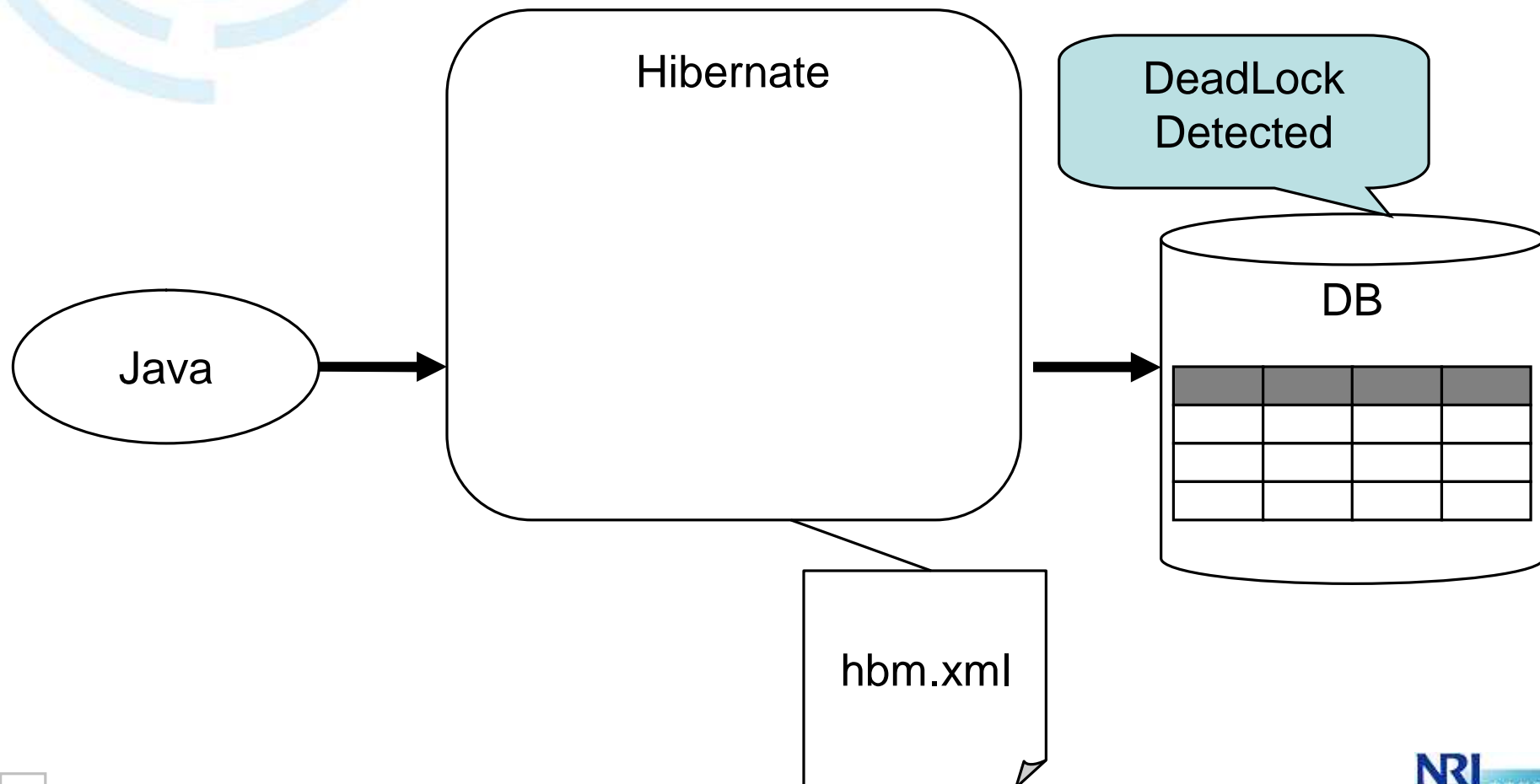
- Hibernateって性能出ないんじゃない？



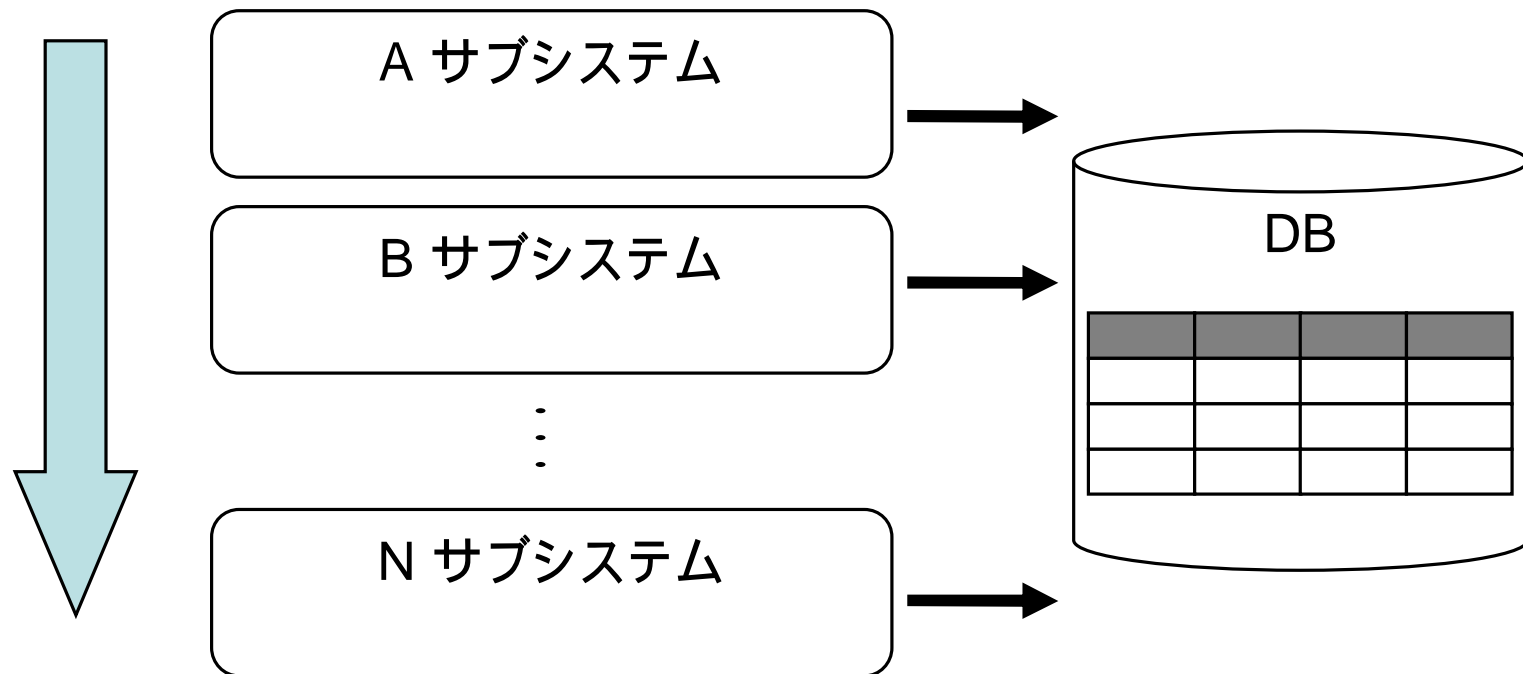
- Hibernateを改修
- コーディングルールを追加



- アプリケーションチームからの問い合わせ
 - デッドロックが心配なんだけど...

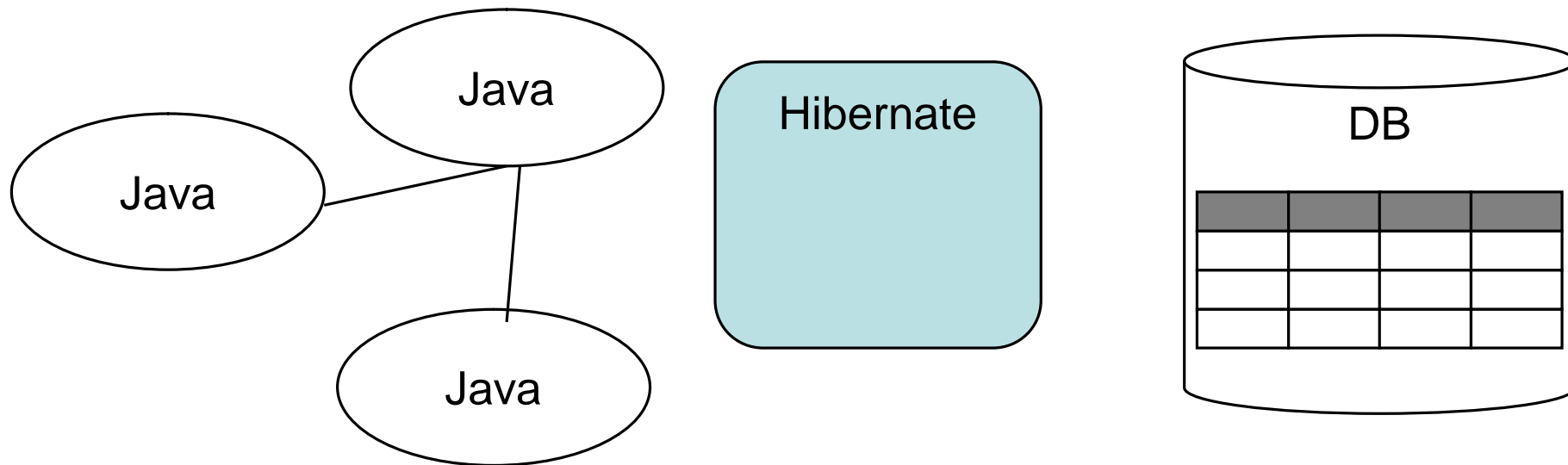


- コーディングルールを追加
 - ロックの順序を定義
 - サブシステムの呼び出し順序を定義

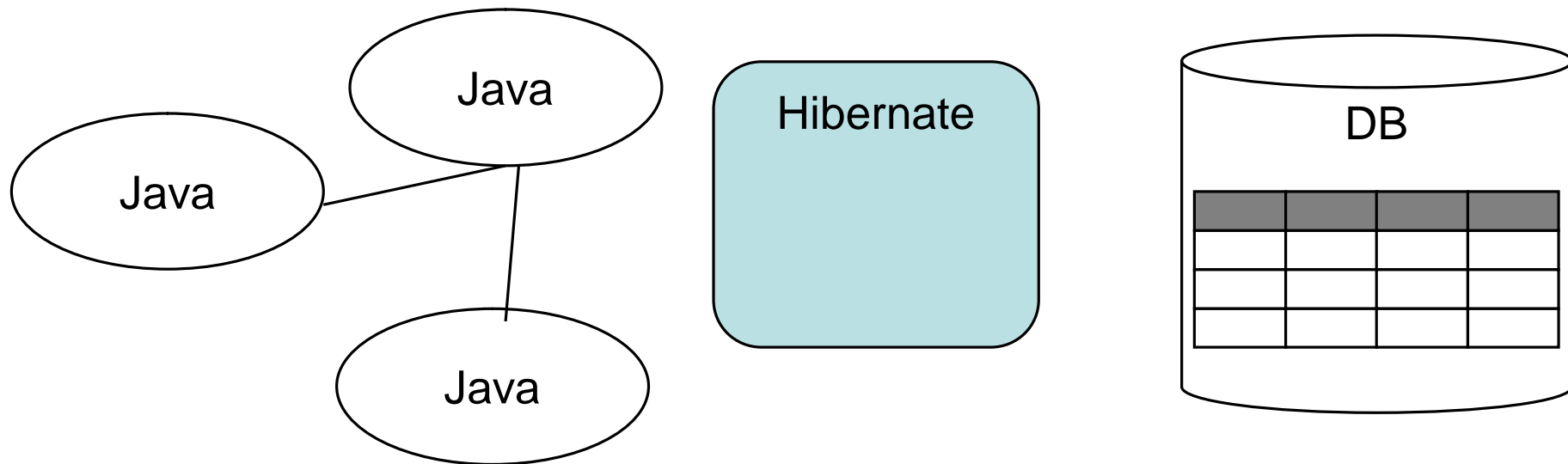


反省点

- Javaオブジェクトをテーブルにマッピングする
- オブジェクトモデリングがちゃんとできないとダメ



- 1リクエストでSQLが数多く発行される
 - Hibernateのキャッシュを生かしていない
- JOINなどによる複雑化、性能劣化
 - ビューの多用



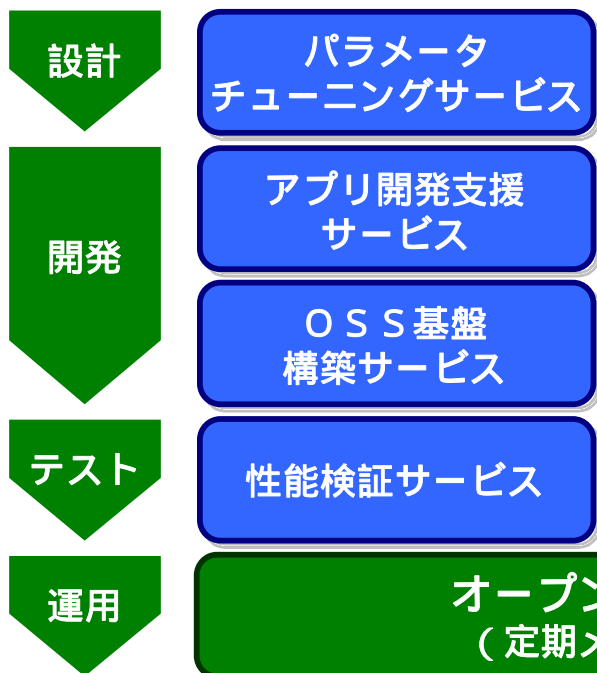
オープンソース・ワンストップサービス OpenStandiaのご紹介

設計・構築から、保守・運用フェーズまで、ワンストップでサポート

オープンソース・ワンストップサービス OpenStandia

新規システム/再構築をお考えのお客様

OSS導入サポート



すでにオープンソースをご利用のお客様

オープンソース 救急センター



OpenStandiaのサポート対象オープンソース

40種類以上のオープンソースを、ワンストップでサポート

CentOS	Fedora Core	Debian	OS
FreeBSD	RedHat Enterprise Linux	その他Linux	
MySQL	PostgreSQL		データベース
Apache	Squid	PHP	Web、スクリプト系
Perl	Ruby on Rails		
Apache Tomcat	Apache Geronimo/WAS CE	iBATIS	Java系
JBoss AS (EAP)	JBoss ESB	JBoss jBPM	
JBoss Cluster	JBoss Seam	Struts	
Spring	Hibernate	Seasar2	
Apache Axis			
Samba	OpenLDAP	BIND	ネット、インフラ系
sendmail	postfix	Dovecot	
CVS	Subversion	Heartbeat	
Eclipse BIRT	JasperReports	osCommerce	アプリ系
OpenOLAP	SugarCRM	Liferay	

WAS CEは「WebSphere Application Server Community Edition」の略です。

まとめ



- お問い合わせは、NRIオープンソースソリューションセンターへ
Mail : osscc@nri.co.jp
Web : <http://openstandia.jp/>