

Company Brochure

Company Brochure

企業データ

会社名	有限会社バーチャルテクノロジー(Virtual Technology, Inc.)
設立	2003年6月13日
事業内容	RESTful SOA、クラウドコンピューティング、並列技術を中心とするシステム 設計・コンサルティング、ミドルウェアの開発やサービスの提供
オフィス	本社 〒814-0033 福岡市早良区有田 8-18-8-105 電話 092-861-5497 支社 〒105-0014 東京都港区芝 2-5-17-1002 電話 03-3451-4179
資本金	300万円 役員 2名 社員 2名 契約社員 2名
代表取締役	竹崎 伸一郎
主な取引先	株式会社 クレスコ 株式会社 NT インフォメーションシステム オーバルテクノロジー 株式会社 有限会社 ミントジャムス アイ・ビート 株式会社 株式会社 特許デイタセンター
お客様・パートナー	株式会社 クレスコ 東京エレクトロン株式会社 日本オラクル株式会社 日本アイビーエム 株式会社 ヤマトシステム開発株式会社 アカマイ株式会社 株式会社 リコー FXIS富士ゼロックス情報システム 株式会社 NSISS 株式会社

事業紹介

信頼をスケールする

Virtual Technologyのクラウドサービス

企業（エンタープライズ）システムがクラウドの上のソフトウェアを利用することになっても、これまでと同じようにデータを普遍的に扱えることが重要です。エンタープライズシステムにおいては、Scalability（規模の変化に対応）やAvailability（可用性）も大事ですが、Consistency（一貫性）、つまり、トランザクション処理が可能なシステムであることが求められます。こういった要求に応えるべく開発されたのがReflexです。

Reflexは、Berkeley DB Java Editionの分散Key/Valueストアを採用しています。

ドキュメント指向であり、データを階層で管理、またJSONやXML、MessagePackといった様々な形式で登録・取得できます。スケールアウト拡張が容易で、かつトランザクション処理ができるのが大きな特長です。

私たちは、このReflexを、トランザクション処理が可能なクラウド基盤として、オンプレミスのPrivateクラウドやAmazon(AWS)といったPublicクラウド環境に提供いたします。



事業紹介

●製品

トランザクショナルKVS ReflexBDB

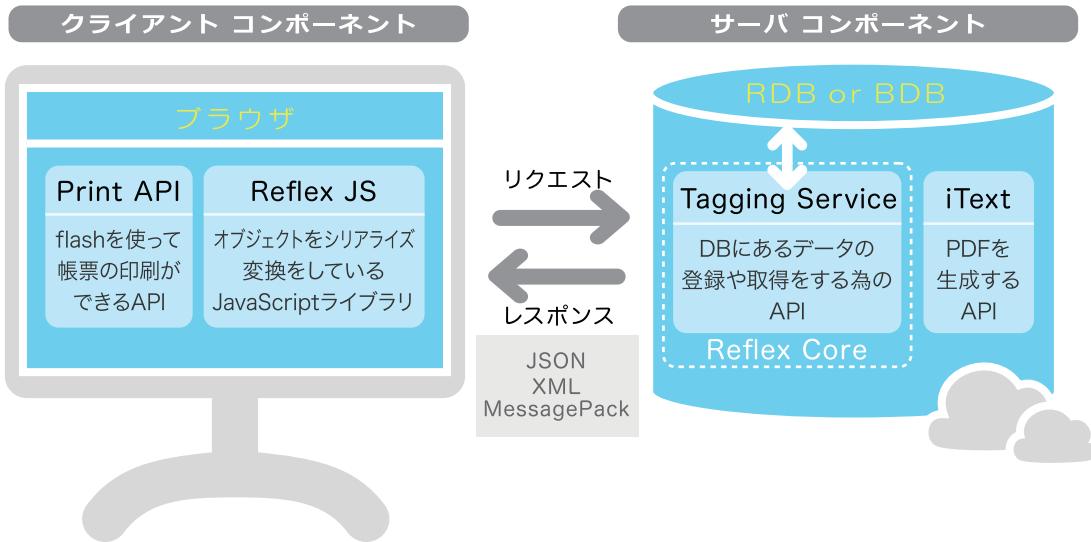
Berkeley DB Java Editionを利用した分散Key/Valueストアです。これは、ハイパフォーマンスなデータアクセス、トランザクションサポート、また、アプリケーションやシステム、ハードウェアで致命的な障害が発生した場合の自動リカバリなどをサポートしています。

Reflex iText

ストリームデータから動的にPDFを生成することができるエンジンです。テンプレートはHTMLとCSSで記述できます。

Flash PrintAPI

Flashの印刷機能を使った帳票印刷用APIです。クライアントのリソースを利用するためサーバに負荷をかけることがありません。



●サービス

コンサルティング

疎結合アーキテクチャーによるWebサービスの設計支援を行います。

プロダクトサポート

Reflex製品に関する技術サポートを行います。

事業紹介

●Reflexの特長



規模の変化に対応 - Scalability -

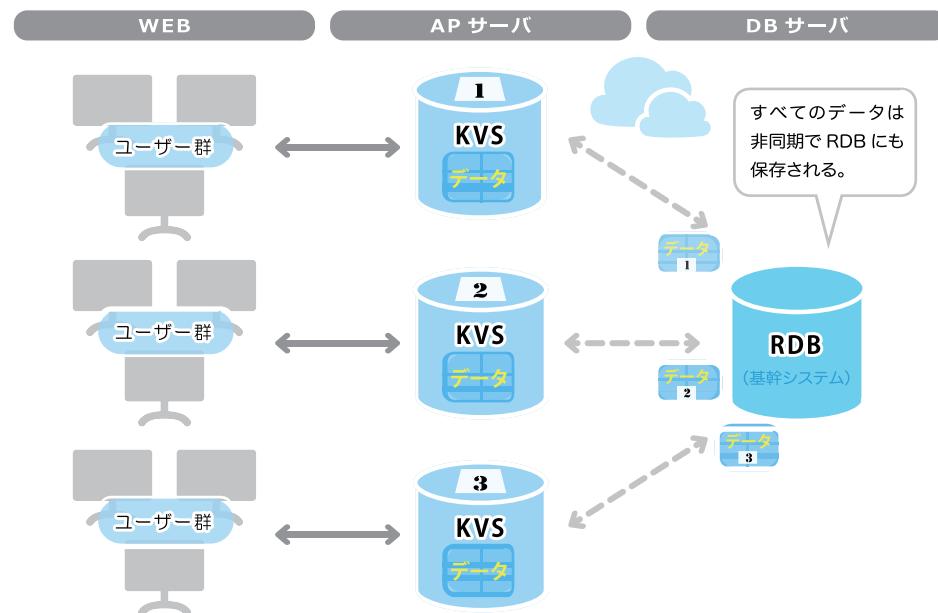
性能や信頼性といった非機能要件を高いサービスレベルで実現するためには、より多くのコストがかってしまいます。アプリケーションを作ってしまった後で性能を向上しようと思っても、高価なハードウェアに置き換えるしか対応できないといったこともあります。

Reflexは、Consistent Hashアルゴリズムを使った分散ノード管理を行っており、スケールアウト・アーキテクチャーのシステムを自社内に構築できます。データは、PartitionIDを元に担当ノードに分散配置されるため、どんなに大量のトランザクションが発生したとしても、単純にノード追加により対応することができます。



ハイパフォーマンス - Performance -

Reflexは、高スループットの待機時間の短い読み取り、高い同時実行性とデータのスケーラビリティを実現するインメモリ・キャッシングをサポートしています。アプリケーションの実行ではRDBアクセスを必要としないため、高いパフォーマンスを実現できます。

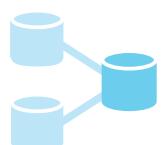


事業紹介



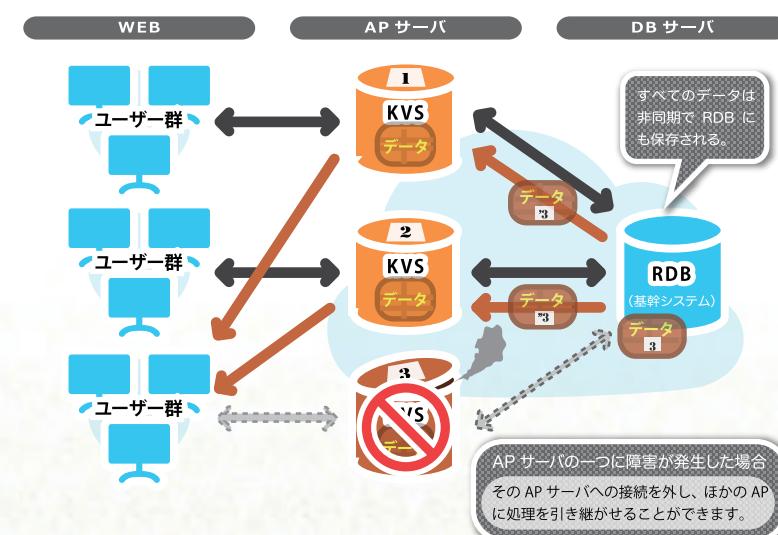
一貫性 - Consistency -

Reflexは、トランザクショナルなKVSである、Berkeley DB Java Editionをデータストアとして採用しているため、トランザクション管理が使えます。つまり、一貫した状態のデータベースを維持するよう設計されており、相互依存のある複数の操作が全て完了するか、全てキャンセルされることを保証します。Reflexはデータの操作をAtom FeedもしくはEntryの単位で実行しますが、一つの論理的の操作が行え、かつ、原子性(Atomicity:不可分性)、一貫性(Consistency)、独立性(Isolation)、および永続性(Durability)といった処理の信頼性を保証します。また、Revision番号管理による楽観的排他機能を利用して並列処理として実行できます。



サービスをいつでも利用できる - Availability -

システムの障害が発生してもサービスが止まらないことは重要です。Reflexは、RDBとのデータ連携を非同期に実行し、その際、APサーバのReflexプロセス（ノード）の監視も行っています。プロセスが何らかの理由で応答しなくなると、自動的にノードリストが書き換えられ、ダウンしたノードのデータが他の正常なノードに引き継がれます。引き継ぐデータは他のノードからではなくRDBから取得します。そのため、複数のノードに重複して保持（レプリケーション）する必要はありません。ちなみに、すべてのノードのデータはRDBに書き出されるため、データ集計などのバッチ業務等は、これまでと同様にRDBを参照することで実行できます。



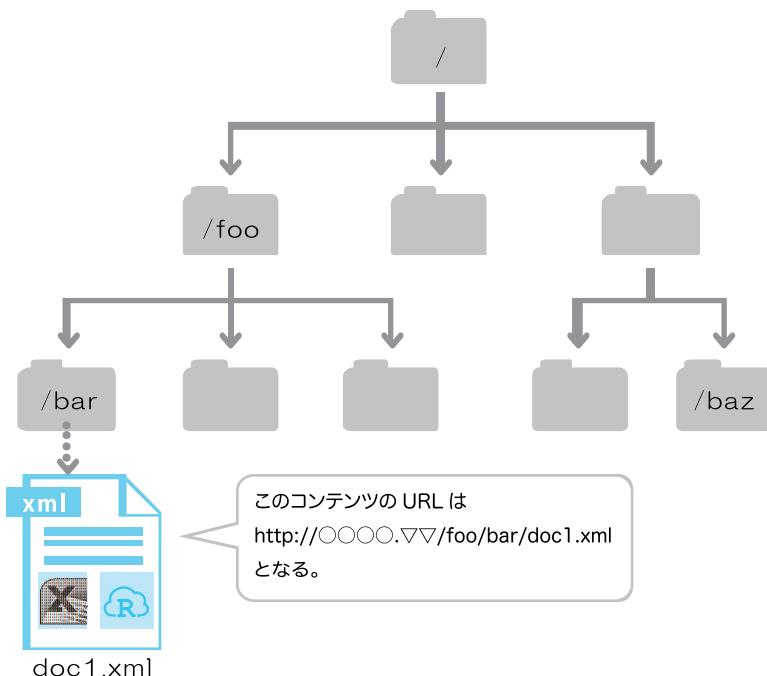
事業紹介



直感的なREST API - API Creation -

Reflexは、ReflexCoreおよび、Tagging Serviceで提供されるAPIを介してデータを操作します。データを階層管理しており、直感的なREST APIによってCRUD操作（作成・読み込み・更新・削除）が可能です。また、様々な条件検索指定や、カウント、自動採番、Pagination、ソートといった機能などもあり、業務アプリケーションの作成を容易にします。ReflexCoreは、JavaオブジェクトからXML、JSON、Messagepackなどに変換するシリアル化機能とXML、JSON、Messagepackをオブジェクトに変換するデシリアル化機能を提供します。Tagging Serviceはデータ操作のためのAPIを、統一したインターフェースとして提供します。基本的に疎結合であり、RDBやKVSなどのデータストアにも依存しないため可搬性を維持できます。APIを介しておくことで様々なクラウドのプラットフォーム上で動作するアプリケーションを容易に作れるようになります。実際にTagging Serviceは、Google App EngineとAmazon EC2のどちらでも動作しますが基本的に同じAPIを利用できます。

仮想フォルダ管理



操作	リソース URL	タイプ
GET	/[フォルダ]/…/[エントリ]	?xml
POST		?json
PUT		?pdf
DELETE		…

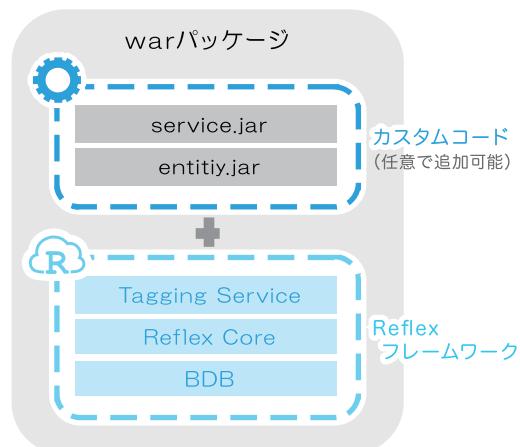
※GETの場合は検索条件を指定可能

事業紹介



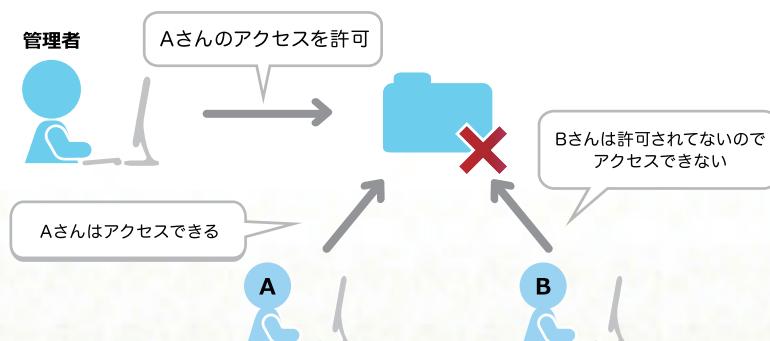
カスタマイズ機能 - Custom Code -

業務アプリケーションにおいてはサーバ側でデータチェックしたいケースがあります。例えば、ECシステムにおける受注処理においては、在庫確認や注文商品の値段などを、サーバの商品マスターを使ってチェックする必要があるでしょう。Reflexでは、クライアントからのREST APIを使ったデータ操作だけではなく、ユーザが作成したビジネスロジックをサービスとして追加することができます。サービスはjarファイルとして簡単にサーバにデプロイすることが可能です。Reflexはアプリケーションで定義するスキーマ（ソフトスキーマ）を採用しており項目の追加変更が容易にできます。スキーマはJavaクラスによって定義し（これをエンティティと呼んでいます）、1つのjarファイルにパッケージしたものをサーバに配置（デプロイ）します。



認証・認可機能 - Auth & Integration -

データを格納するフォルダにはアクセス権限(ACL)をつけることができます。アクセス権限が付いたフォルダの配下のデータは、ログイン済の許可されたユーザのみがアクセスできます。ACLは基本的には許可ユーザを個々に指定しますが、ワイルドカードが記述できるため、グルーピング指定も可能です。Reflexには独自のユーザ管理機能がありますが、TwitterやFacebookといった外部のOAuth Providerとの連携も可能です。認証では、リピート攻撃、総当たり攻撃（含む辞書攻撃）に対応するため、ワンタイム認証やRevoke機能（複数回認証失敗でパスワードリセット）、また、IP Blacklistによるアクセス防止機能などを備えています。Reflexは、項目ごとに暗号化指定が可能であり、万一、外部にBDBファイルが漏洩した場合にも解読されることはありません。

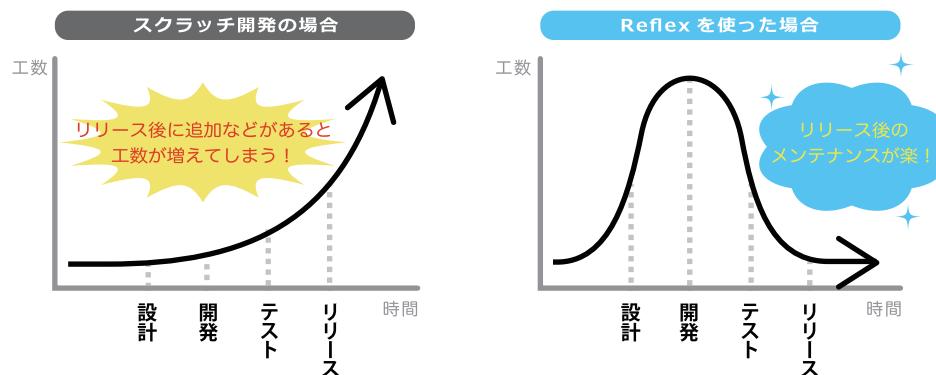


事業紹介



品質と生産性 - Quality & Productivity -

技術要件（システム実現方法や開発方式など）が一貫していないと高い品質を維持するのが難しくなります。たいていは汎用性のない作りでリリースしてしまい、後にメンテナンスや機能追加をする作業が大変になってしまいます。Reflexは、統一したエンティティモデルを採用しているため、DB、サービス、画面で一貫した項目名が使われます。

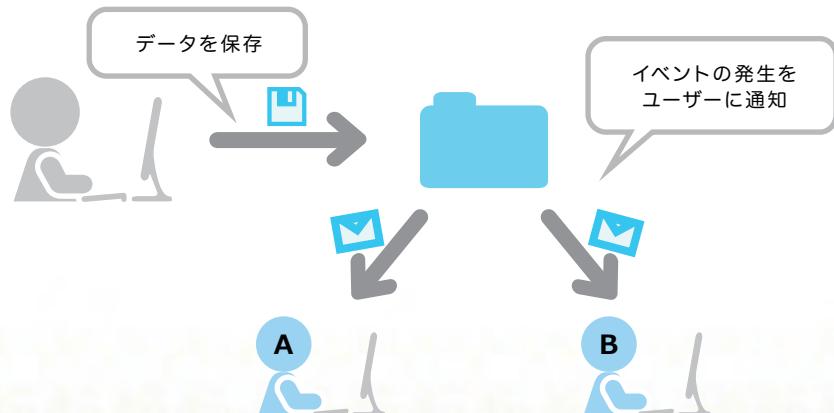


BDBを使う場合においては、オブジェクトが直接格納されるため、ORM(O/R マッパー)を必要としません。その結果、項目変換処理はほとんど存在せず、同意語(synonym)が発生しにくいという利点をもちます。このため、システム全体として統一感のある品質の高いシステムができあがります。

イベント通知機能 - Push Notifications -



ソーシャルなアプリケーションでは、更新通知機能が欲しくなるケースがあります。例えば、他のだれかの発信情報をリアルタイムで見たり、データが更新されたことをすぐに知りたい場合などです。Reflexでは、あらかじめフォルダにアクションをセットしておくことで、フォルダ配下のデータにアクセスしたタイミングでイベントを通知することができます。イベントが発生すると、WebHookにより外部システムへ通知、あるいは、WebSocketによりブラウザへ通知されます。これらの機能により、インターラクティブなアプリケーションは容易に作れるようになります。



事業紹介

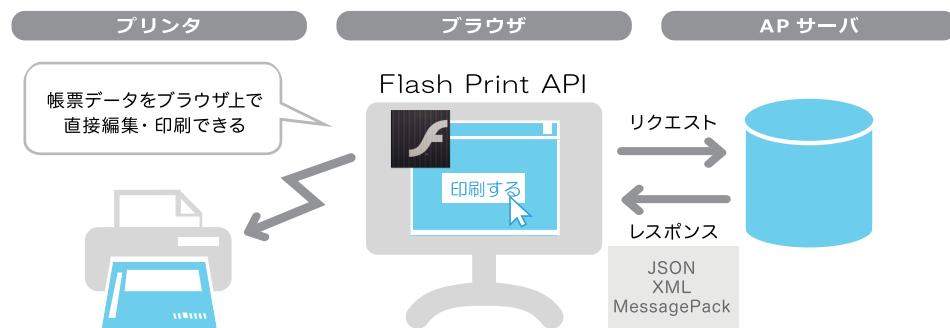


帳票印刷 - Printing -

帳票生成処理はCPU負荷が高くメモリも多く消費します。業務アプリケーションには必ずといっていいほど帳票出力があり、それがサーバ資源を圧迫させています。Reflexでは、クライアントから直接印刷するFlash PrintAPIと複数のサーバを使ってPDFを生成するReflex iTextの2つの方式を提供します。

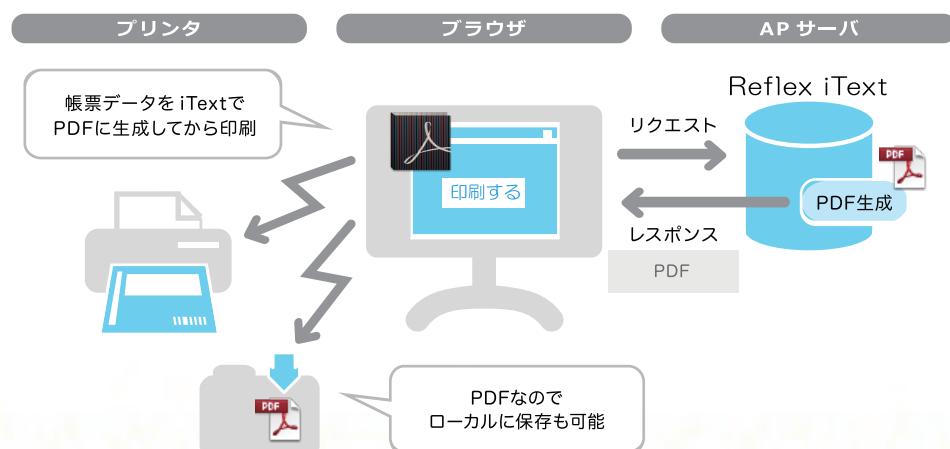
Flash PrintAPIは、サーバで帳票を作成するのではなく、FlashのPrintAPIを使って印刷するもので、基本的にクライアントのリソースだけを使うためサーバ資源を圧迫せることはあります。

パターン1：ダイレクト印刷



Reflex iTextは、Amazon EC2のインスタンスを動的に複数起動させ、帳票レイアウト(HTML)とデータを動的に与えることで大規模なPDF生成処理を可能にします。アプリケーションサーバとEC2のインスタンスは基本的に疎結合でありアプリケーションサーバの資源を圧迫させる心配がありません。

パターン2：PDFにして印刷



沿革

2012	大手宅配業者 送り状発行システム、サービスイン Reflex BDBによるスケールアウトシステム、Flash PrintAPIによる帳票印刷
2011	Flash PrintAPIを発表 Flashの印刷機能を利用した帳票ソリューション
2010	Reflex Tagging Service発表 KVSで直感的にデータを扱うためのREST API
2009/05	Scalable PDF 発表 Reflex iText を Amazon EC2 上で実行して動的にインスタンスを増減できる
2009/05	Reflex Core for Google App Engine 発表 Reflex Core の Google App Engine 対応。GAE から JSON や ATOM、SOAP 通信が可能に
2009/01	Reflex BDB 発表 Key/Value ストア Reflex BDB の発表
2008/12	クラウド PDF サービス Reflex iText on Morph App Space 発表 Morph と DropBox でクラウド PDF サービスを実現
2008/11～2009/03	Reflex iText でアカマイ様とのアライアンス提携 Reflex iText を Akamai Edge Computing に対応
2008/11～2009/03	某大手通信キャリア向けアプリケーション SNMPforJava などを使ったアプリケーション開発
2008/07～2009/01	SAS データ連携システム EC システムと SAS システムとの連携アプリケーション (PHP)
2008/09	大手ディーラー向け請求書発行システム Reflex iText による帳票システム
2008/04～2008/08	損益計算システムの開発 Reflex View によるブラウザベースの高速アプリケーション
2007/01～2008/06	Web2.0 EC サイト開発 暮らしのデザイン EC サイト
2006/11～2006/12	Adaptor システム機能評価技術支援 SCA 製品の機能評価
2006/10～2006/11	某大手自動車ディーラー向け 譲渡証明書発行システム開発 Reflex iText を使った PDF 帳票システム
2006/4～2006/12	某大手 IT 企業向け オブジェクト指向設計標準化活動支援 DI コンテナ、AOP をベースにした新アプリ基盤構築
2006/5～2006/7	某大手自動車ディーラー向け 販売活動分析システム開発 AJAX を使った棒グラフ、折れ線グラフ表示
2006/5～2006/6	某大手 OA 機器メーカー向け 処方箋証明プロトタイプ開発 Scanner を使ったデモシステム開発
2006/1～2006/3	某大手 OA 機器メーカー向け Web2.0 プロトタイプ開発 AJAX、ATOM PP を使ったデモシステム開発
2005/6～2005/10	XML コンソーシアム 愛・地球博用デモ開発支援 地図と道路交通情報のマッシュアップが特徴
2005/7～2005/12	某大手 OA 機器メーカー ドキュメント管理システム機能改善 リッチクライアント (WCTME) 技術採用
2005/5～2005/6	某大手石油会社、試験成績書発行システム設計開発 Struts、iText による開発
2004/7～2005/4	某大手 OA 機器メーカー ドキュメント管理システム設計開発支援 SOA メソドロジーによる開発、SOAP (DIME)、リッチクライアント (WCTME) 技術採用
2004/3～2004/6	XML コンソーシアム TravelXML デモ開発支援 IBM (jStart) として TravelXML デモ開発に参加
2004/2～2004/5	Turbo Linux 様の Media Player 開発支援 Turbo Linux 10F にバンドルする Turbo Media Player を開発
2004/1～2004/3	FXIS 様の SW キー発行システム構築支援 Linux、Tomcat に NetVault、LifeKeeper を組み合わせた高可用性システムを構築
2003/9～2003/12	RTS(リコーエクノシステムズ) 様 MaBot の設計支援 SOAP と UDDI を使った業界初の Web サービスシステム。IBM jStart の事例
2003/9～2003/12	FXIS 様の人材管理システム開発支援 Struts ベースのアプリケーション開発
2003/8～2003/10	某大手自動車ディーラーの営業支援システムの開発支援 iSeries(AS/400) の WebSphere と UDB を使用

拠点アクセス

本社

〒814-0033 福岡市早良区有田8-18-8-105

支社

〒105-0014 東京都港区芝 2-5-17-1002 TEL : 03-3451-4179



東京メトロ 大江戸線「赤羽橋」徒歩7分
三田線「芝公園」徒歩5分