

Part 4

事例①:コニカミノルタフォトイメージング株式会社「オンラインラボ」

画像データがメインの商用サイトをオープンソースのみで構築

本パートでは、コニカミノルタフォトイメージングの「オンラインラボ」サービスにおける、PostgreSQLの活用例を取り上げる。オンラインラボは、デジタル写真のプリントのほか、大量の画像を扱うオンラインアルバムなどのサービスを提供し、それらのシステムをすべてオープンソースで実現している。ここでは同社のオープンソースへの取り組みと5年以上にわたる活用の実績を紹介する。

株式会社SRA
稲葉香理
INABA, Kaori

コニカミノルタフォトイメージング(以下、コニカミノルタ)では、インターネットからデジタルデータのプリントを注文できる「オンラインラボ」システムを2001年から運用している(<http://onlinelab.jp/>)。

ハードウェアの増強やバージョンアップなどをくり返しつつも、オープンソースでのシステム構築ポリシーを変更せずに現在まで運用を続けている。本稿では、このシステムの現状と、オープンソースを用いたシステム構築にいたるまでの経緯、開発や運用において苦労した(している)点をまとめ、同サービスの今後のシステム展開などについても解説する。

オンラインラボシステムの概要



「オンラインラボ」は、デジタル写真に関する各種のサービスを提供するBtoCサイトである(画面1)。同サイトでは、アップロードしたデジタル写真を印画紙に焼きつけユーザーに送付する「オンラインプリント注文」、暑中見舞いや年賀状などの「ポストカード注文」、マグカップやTシャツなどに写

真をプリントできる「オリジナルグッズコレクション」、会員ユーザーがインターネット上に写真を保存できる「オンラインアルバム(画面2)」など、幅広いサービスを提供している。

システム全体は、「センターサーバー」と呼ばれるインターネットに直結するシステムと、各店舗システムから成り立つ。一般ユーザーから見られるのはセンターサーバー部分である。

センターサーバーは、多数のWebサーバー、メールサーバー、NFSサーバー、そしてデータベースサーバーから成る。センターサーバーのシステム構成は図1に示す。1日のアクセス平均は約10万ページビュー、繁忙期(キャンペーン/年末など)には50万ページビューに達する。また、オンラインプリントの注文数は毎日相当数に上っている。

店舗システムはコニカミノルタと提携している写真店用のシステムで、店舗で受け付けた注文をセンターサーバーに送信する。

データベース

同サイトの利用ユーザーは、次の3者で

定義されている。

一般ユーザー:インターネットからの注文やサービスを利用する

店舗:いわゆる街の写真店で、注文のあったプリントを依頼する

管理者:コニカミノルタ

データベースにはPostgreSQL 7.3.2を採用し、以下の構成で運用している。

OS: TurboLinux(カーネル2.4.21-2smp)

CPU: AMD Opteron 2GHz×2

メモリ: 4GB

このほか、オンラインラボでは、冗長性を確保するためのホットスタンバイ構成を導入している。本番用サーバーと同じ構成の代替サーバーを準備し、本稼動サーバーに何らかの障害が生じた場合は、RAID構成のHDDをそのまま移し替えて復旧する。仮に、RAID自体が故障するという事態が生じた場合でも、3時間ごとに取得しているpg_dumpall(PostgreSQL)のデータを代替サーバーに適用し、その後はアプリケーションサーバー



画面1 オンラインラボ



画面2 公開アルバム

が記録しているトランザクションをDBに適用することで復旧できるようになっている。データベースに格納している情報は、登録会員の情報、オンラインプリントの注文情報などを中心に約200のテーブル、容量は15GB程度で、1000万近い画像の管理を行なっている。

データベースのアクセスは、検索>登録>更新>削除の順が多い。

ただし、上の構成にはオンラインアルバム用のデータは含まれない。オンラインアルバム用のデータはデータベースへの収納を避け、約6TBの画像データを別のストレージで管理している。

接続数の設定管理

サービス開始当初は、Apacheのデフォルト値である150を接続数上限とし、Webサーバー以外からもアクセスのあるDBサーバーについては256とした。

その後Webサーバーはアプライアンスを用いて多重化し、同時にHTTPのコネクションを絞り込むことで、DBサーバーの設定を変えずに運用を続けている。現在はマニカのWeb I/Oアクセラレータ「TIX 2000」のSNMP機能でデータを収集し、

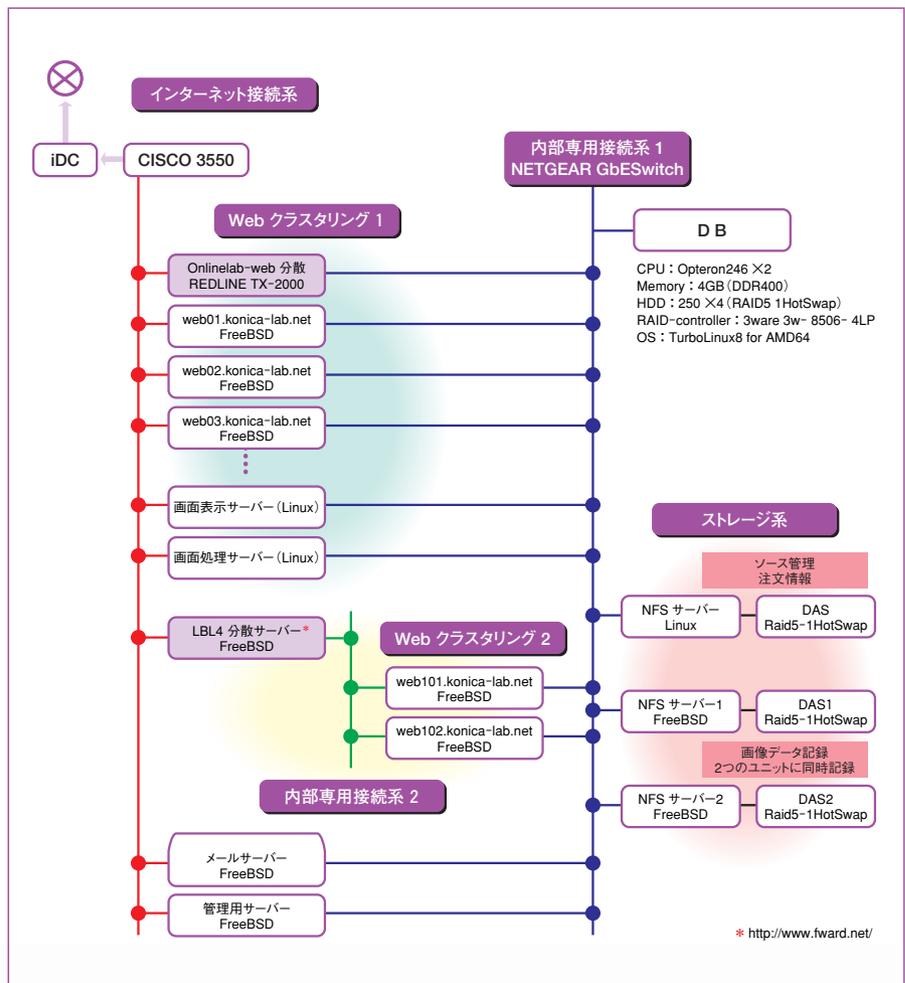


図1 サーバー構成図

* <http://www.fward.net/>

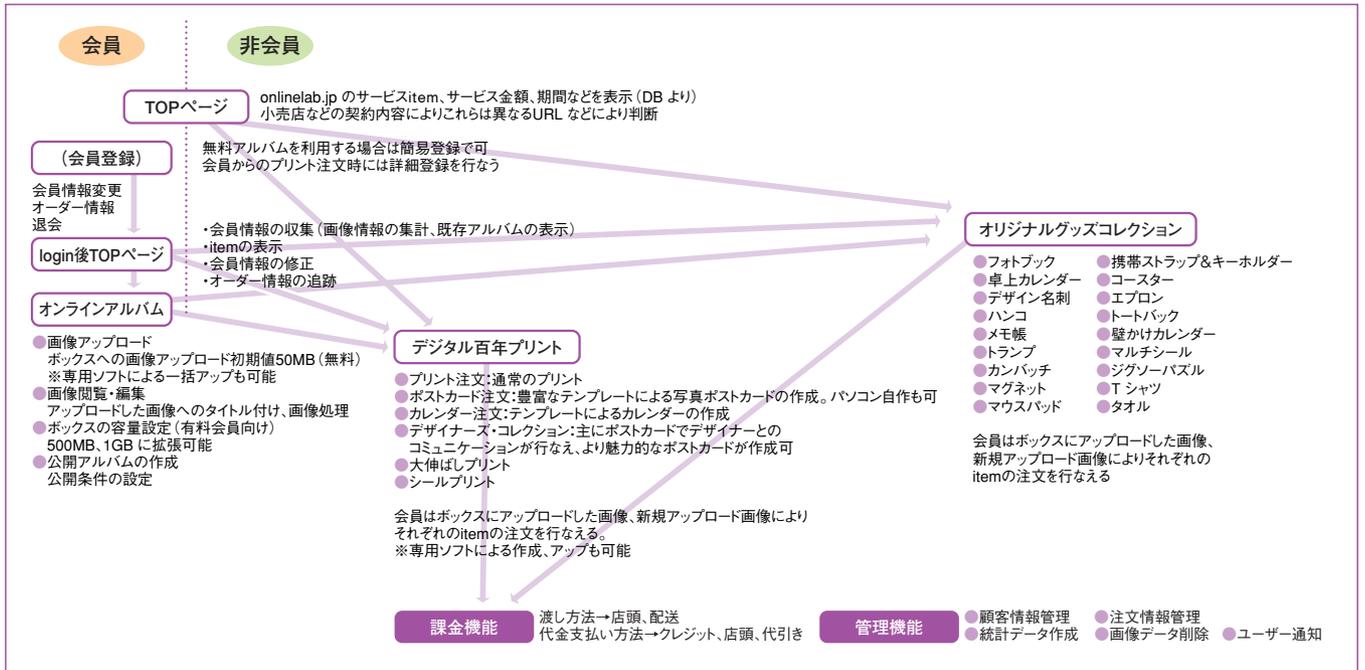


図2 サービス構成図

MRTG(Multi Router Traffic Grapher)でコネクション数やプロセス数を監視しながら、設定値を調整している。

注文のステータス確認に関しては、センターサーバーのDBにステータス情報を格納し、同一情報を顧客は会員画面、店舗は店舗の注文管理画面で参照できる(図2)。注文のステータス遷移は、センターサーバーと現像所サーバーの間でやりとりがなされ、現像所での注文の処理状況がセンターサーバーのDBに反映されている。



コニカミノルタフォトイメージング 飯塚宏之氏

オープンソース 利用の経緯



NEDOへの応募

オンラインラボ事業においては、コニカミノルタ・イメージコミュニケーション事業部の飯塚宏之氏(写真)が中心的な役割を果たしている。同氏は以前から写真の発色に関する量子化学のコンピュータシミュレーションを研究していたが、その過程で国内の研究期間や大学で利用されるインターネットの世界に興味を持ち、オープンソースの機能性や安定性を実感していた。

コニカミノルタでは、1995年頃から社内でのメールサーバーやWebサーバーをLinuxで立ち上げ、1999年には、NEDO(独立行政法人産業技術総合開発機構)の公募に参加し、FreeBSD + PostgreSQL + PHPで構築したインターネット上の写真関連システムのプロトタイプを完成させていた。現在のオンラインラボは、このシステムを構築した経験を元に作られている。

ビジネス展開に向けて

インターネットを活用したビジネス展開は、1999年ごろから本格検討が進められた。当初はホスト系システムでの構築も検討されたが、多大な費用が必要との判断から、オープンソースによるシステムによる構築方針が定まった。

ただし、アプリケーション開発まで自社で賄うのは難しいとの判断から、開発に関してはSRAに委託する方針がとられた。ちょうどこの時期、SRAはオープンソースを専門に扱うサービスを立ち上げており、特に同社の(当時日本PostgreSQLユーザー会理事長であった)石井達夫氏を中心にPostgreSQLに関するノウハウを蓄積していた。

こうした体制の整備を進めた結果、OSはFreeBSD(現在は一部Linuxも併用)、WebにApache、アプリケーションの開発にPHPを利用し、RDBMSにPostgreSQLを採用したシステム構成が決定し、そしてこのシステムは現在まで運用を継続している。

開発と運用のポイント



オープンソースのみで構成されたシステムの利点と苦勞した点について、飯塚氏は次のように話す。

「オープンソースについては、最初はコスト的なメリットが大きかったのですが、最近ではセキュリティの堅牢さを特に評価しています。ソフトウェアのアップデートは頻繁で、セキュリティホールが見つかって、すぐにパグフィックスが行なわれる点は、最近ではとくに重要になってきていると思います。逆に、苦勞した点は、当初は、サーバー構築からテストまでをすべて自力で解決しなければならなかった点でしたが、最近ではSRAのような企業からオープンソースに関するサポートも提供され、サービスを拡大していく際にも安心して使えるようになってきています。システム開発時は、特にデータベースの設計に留意し、SRAのコンサルティングを受けながらER図を用いたデータ中心手法を用いています」

開発時は設計が特に重要

飯塚氏に設計時のポイントなどを聞いた。「写真業界は歴史の長い業界で、ビジネス展開も多岐に渡り複雑です。インターネットビジネスと言うと、流通の中間段階を省略することによるコストダウンというキーワードがすぐに思い出されますが、弊社にとっては婚礼や子供の記念写真のご商売をなさるプロ写真館や、DPEショップがエンドユーザー様と同じように大切です。

したがって、DB設計では通常のインターネットビジネスで行なわれるエンドユーザーダイレクトのビジネスプロセスとともに、従来の商習慣に基づく商流／物流を表現できるように配慮しました。

具体的に悩んだのは、商品を作成する手順をデータベース設計に反映する部分です。商品がそのまま売れて終わりではなく、撮影

表 データベースサーバーの変遷と増強

	CPU	OS	RDBMS
2001年6月	Pentium III×2	FreeBSD	PostgreSQL 7.0
	Pentium 4×1	FreeBSD	PostgreSQL 7.0
	Xeon×1	FreeBSD	PostgreSQL 7.2
	Xeon×2	FreeBSD	PostgreSQL 7.2
2004年	Opteron×2	TurboLinux	PostgreSQL 7.3

した画像を最終商品としてユーザーにお渡しするまでに複数の手順を踏むため、1つの商品に対して、焼く(記録する)画像とそれを乗せる媒体、同時に記録するキャラクターや文字データなど複数の要素をDBで管理できるようにしました。

設計にあたっては、各要素をできるだけ抽象化し、印画紙や葉書を『媒体種』として、一緒に記録するものを『テンプレート』としてテーブル化する工夫をしました。しかし、ビジネスの発展にともなって、グッズなどの新商品を追加して行く際に従来の抽象化では記述しきれなかった面もあります。このため、新しい媒体のためにテーブルを起こして対応することもありました」

WALの効果を実感

DBに対しては、キャンペーンの開催や年賀状／暑中見舞などの締め切り時に大きな負荷がかかる。また、会員数の増加にも対応するため、1999年のサービス開始時からハードウェアのリプレース、もしくはPostgreSQLのバージョンアップを続けている(表)。

基本的に、1台のDBサーバーでの運用を前提としているため、サーバーの性能をいかに引き上げるかがポイントになる。

PostgreSQL 7.0から7.2へのバージョンアップは、7.1から実装されていたWAL(Write Ahead Logging)によるパフォーマンスの向上を目的としていた。

7.1以前のPostgreSQLにはディスクアクセスを低減する非同期モード(fsyc = off)があったが、障害時に一部のデータが失われる可能性があったため、同期モード(fsyc = on)を利用するのが一般的だった。これに対し、WALでは先行してログファイルに書き込

みを行ない、実データの書き込みは後から行なうため、データの完全性を確保しながらも、ディスクへの書き込み時間を大幅に減らすことができる。

また、2004年のOpteronの導入は、意外なほど効果があったという。PostgreSQLが64ビットOSで動くことは案外知られていないが、性能的なメリットはかなり見出せるとのことである。

システム安定稼働のためのTips



以上のように、コニカミノルタでは、オープンソースを活用したシステムを構築し、約5年にわたる長期の稼働を実現した。しかし、この5年の間には、運用に関する苦勞も数多くあったという。以降では、そうしたオンラインラボの運用経験から、いくつかのエピソードをピックアップしてTipsとして紹介したい。

バージョンアップでの苦い経験

本稿執筆時のPostgreSQLの最新バージョンは7.4だが、オンラインラボでは常に最新バージョンを使用する方針はとっていない。これは、PostgreSQL 7.2へのバージョンアップ時に、思いのほか大きなコストを生じた経験があるためだ。

PostgreSQLは、pg_dumpコマンドによる各バージョン間のデータの互換性を有している。しかし、オンラインラボにおける移行では一部のデータに不整合を生じ、スクリプトや手作業による修正を余儀なくされた。この影響で、移行作業が定期メンテナンスの時間内に収まらず、何度か試行を繰

り返すことになったほか、アプリケーションの大幅な改変も生じた。

このケースでは、メジャーバージョン間の移行であったこと、データ量の多さなどの制約があったが、データベースのバージョンアップは、機能や性能向上に対する期待と、それににかかる工数とのトレードオフであることが分かる。

常に安定している新しいバージョンを使うことは、セキュリティホールの発見などについては有利だが、必要性がない部分に対しては、無理にバージョンアップを行わない方針もあり得るということである。

データ領域の再構築

PostgreSQLは追記型のデータベースであるため、運用時にはvacuumコマンドを実行して、データ領域の再構築を行う必要がある。PostgreSQL 7.2では運用中にも実行できるconcurrent vacuumが利用可能になっており、オンラインラボでは、これとvacuum fullを併用する方針をとっている。

concurrent vacuumは不要データ領域の再利用を可能にするコマンドであるが、1日1回ペースでconcurrent vacuumを実行し、週に1回の定期メンテナンス時にvacuum fullを実行している。vacuum full

ユーザーによるトラブル

技術的なトラブルのほか、悪意のあるユーザーによるトラブルもある。写真をアップロードしているようにみせかけ、プログラムファイルをアップロードし、不法にバラまくといったことだ。ネットワークのトラフィックの集中から発見され、すばやく対応したため、事無きを得たと言うが、こういった悪意あるユーザーの利用に対する防御は、いろいろな観点からのアプローチが必要だろう。

については不要領域を削除するために高レベルのロックを利用しており、運用中の実施には適さない。

画像のアップロード

写真のファイルのアップロードにはPHPの機能を利用している。この機能を使えば、簡単にファイルをアップロードできるが、アップロードするファイルがいったんすべてメモリ上に展開されるため、複数ユーザーが同時にアップロードを行なうとメモリ空領域が不足する場合がある。オンラインラボでは、Webサーバーの分散で対応する方針をとった。

トラブル対応

オンラインラボの運用は、コニカミノルタのスタッフが直接行っている。これは、運用方針として次のような制約があるためである。

- 週1回のメンテナンス時間以外の連続運用
- ダウンタイムは最大でも1時間以内

「ハードウェア／ソフトウェアトラブルも含めて1時間以内のダウンタイム」という条件で外部に運用を任せると、相当なコストを生じてしまう。オンラインラボでは、システムにトラブルが起こるとスタッフの携帯にメールが届くようになっており、トラブルにはスタッフ自ら対応するという。

また、メーカー製のサーバーに頼らず、独自仕様の信頼できるハードを複数用意しておく方針をとっている。サーバーやパーツは常に予備を用意し、画像サーバーなどは、アクセスも多く稼働率が高いので、保管を二重化すると同時にリダンダントや冷却ファンなどの工夫もしている。

トラブルの件数として一番多いのはハー

ドウェア障害である。やはり、使っていれば必ず壊れることを実感するようだ。最近は、フェイルオーバーなどの機能により、自動的にサービスが切り換わるため、初期のころほど人が対応することは少なくなった。

今後のシステム展開



オンラインラボは今後も運用を継続し、順次システムの増強などを行なう予定である。

データベースに関しては1台のサーバーで、性能や可用性をどのように上げていくかが課題である。PostgreSQL自体も、大規模なシステムで使われるようになっており、レプリケーションや、可用性を高めるソリューションが注目されている。

PostgreSQLには、いわゆる標準のレプリケーションソリューションはないが、飯塚氏は特に「PGCluster」に注目しているという。PGClusterは、PostgreSQL用のマルチマスタのレプリケーションシステムである。挿入や更新は遅くなるものの検索の負荷分散が行なえる点、複数のデータベースの内の1台が壊れても自動的に切り離し、オンラインでそのリカバリができる点などを考慮している。同社では、すでにPGClusterをベースに商用サポートを添付したSRAのPowerGres Clusterをテスト導入しており、近々本番システムでも利用される予定である。

DBM

稲葉香理 (いなばかおり)

株式会社SRA勤務。1999年のPostgreSQLビジネス設立時からのメンバー。PostgreSQLサポート、トレーニング、PowerGres開発などを担当している。

E-Mail : i-kaori@sra.co.jp