

IT21 の会(平成 20 年 12 月)第 126 回例会議事録

日 時:2008 年 12 月 5 日(金) 18 時 30 分-20 時 20 分

出席者:18 名(講師含む)

場 所:葺手第 2 ビル 5 階

配布資料:

- 0812-0 (社)日本技術士会 CPD 行事参加票
- 0812-1 基礎から学ぶ Web アプリケーションの性能設計と性能テスト
- ※回覧資料 コミュニティ活動に関するインタビュー調査へのご協力をお願い
(みずほ情報総研)

議事

1. 役員からの連絡事項(会長 加納幸博氏)

(1) コミュニティ活動に関するインタビュー調査への対応の件

- IPA からの委託により、みずほ総研が「専門家コミュニティに関する実態調査」を実施している。このインタビューを約 1 時間ほど受けた。内容は人材育成に関するものが中心であったが、当会としては人材育成自体を直接の目標としているものではない旨説明した。また、本インタビューは PEAK/IT、日本システムアナリスト協会なども取材する予定とのこと。

(2) 半日合宿の件

「(仮称)プロマネ異業種バトル」を開催する計画があり、6 名程募集中。

(3) 合格者向けの PR パンフレット作成について

来年の合格者歓迎会向けの当会の PR 用として、パンフレットへ新たに新コンテンツである SNS、英語版 HP 及び会員出版紹介の追加を計画している。現在広報委員へ依頼中。

(4) 会費の徴収状況について

現在、会費の徴収率は約50%程度にとどまっている。対策として、払わない人は継続の意思なし とみなして一度退会してもらい、支払時に再度入会という方法も検討しなければならないと考えている。

2. 講演 山下真吾氏

- 自己紹介:7年前に独立し、現在、企業の講師、ITコンサルタント、IT関連の書籍の執筆等に従事している。今年、技術士2次試験に合格した。

(1)目次

- システム基盤方式設計(特に性能設計)の進め方
- Web アプリケーションの内部構造
- リソースの見積方法(CPU数、メモリサイズ等)
- 性能テストの計画と実施

(2)内容

システム基盤方式設計(性能設計)の進め方

- はじめにシステム基盤の要件分析を行う。性能にかかわるのは、「非機能要件」の部分である。
- 性能は具体的には、「スループット」「レスポンスタイム」と呼ばれる数値のことを指す。この数値は要件分析の工程で目標値を決める。
- 業務分析の中で、以下の項目を検討しておく。
 - システム滞留時間、単位時間当たりのアクセス数、アクセス毎の要求／応答データサイズ、単位時間当たりのセッション数、セッション毎のセッションデータサイズ …(※)
- 次に行う方式設計では、「システム基盤アーキテクチャ設計」の中に「性能設計」が置かれる。この時点では、CPU数やメモリーサイズは、概算値を求めておく。

システム基盤基本設計(性能設計)の進め方

- はじめに性能目標を明確化する。上記(※)の項目について、一日の平均値とピーク時間帯について目標数値を決定する。
- つぎに、性能分析対象業務の選定を行う。C、C++、などでメモリを任意の時点で解放する場合とJavaなどでガベージコレクションを実施する場合とは、

少しパラメータの取り方が異なる。Javaの場合、ガベージコレクションを実施しないと、オブジェクトが使用済みであっても、ヒープ領域を消費し続けるので、メモリの無駄となる。

【参考】Webアプリケーションの内部構造

- Webアプリケーションの構造は、MVCモデルと呼ばれる。MVCとは、モデル、ビュー、コントローラの英語の頭文字である。コントローラは、Webブラウザからのリクエストを受信し、以降の処理を制御する。モデルはコントローラから受け取った特定のビジネスロジックを処理する。コントローラがビューに処理結果を渡すと、ビューは画面を生成し、ブラウザに返信するという仕組みである。
- 性能を考慮した設計としては、「同時リクエスト数がボトルネックにならないようにする」「静的コンテンツがキャッシュされるようにする」などがある。

【参考】GC(ガベージコレクション)のしくみ

- GCはメモリを有効活用をするための手法であり、性能に与える影響が大きい。GCの対象となるのはメモリ領域のうちJavaアプリケーションが使用するJavaヒープである。
- Javaヒープ領域は、NEW領域とOLD領域に分かれ、さらにそれが細分化されている。GCはこれを無駄のないように入れ替えを行うが、これにはCopyGCとFullGCがある。

リソース(CPU数、メモリサイズ)の見積り方法

- はじめにサーバや回線などのリソースを洗い出し、次にリソースへのアクセスやトラフィック量を把握する。最後にワークロードとして記述する。ワークロードのまとめ方には、リソース使用マトリックス(CRUD分析など)やトラフィック分析表などがある。
- 次に性能確保のための適用技術を決める。たとえば、リバースプロキシ、キャッシュサーバ、負荷分散装置などがある。
- これらのモデルをもとに予測を立てる。予測の方法には、ベンダーの公表値、公式や経験値などを活用する。これは開発の初期段階から予測する方法を考慮することが望ましい。
- CPU数、メモリサイズはキャパシティ設計の中で算出する。なおサーバはリバースプロキシサーバ、DBサーバ、APサーバなどそれぞれに計算する。

- 性能向上策を紹介した『高速サイトを実現する14のルール』という本があり、ヤフーの社員が書いたもので、実践的である。

性能テストの計画と実施

- 性能テストは、『負荷テスト』と『限界テスト』に分けられる。これらは明確に区別して、考える必要がある。
- 負荷テストはプロット開発時、システム開発時、開発後それぞれに目的が異なる。限界テストは、主に、リソース競合の極限状態に陥った場合でもシステムダウンに至らないかを検証する。
- 負荷テストにあたっては、目標スループットが要件定義で定まっていることが前提である。負荷テストは同時実行ユーザ数の最大値を求めるが、その途中のユーザ数も求めていく方法をとる。
- 限界テストはピーク時を大幅に上回る大量の負荷をかけ続ける。この際、キャッシュサーバを通さずに使用するなどの注意が必要である。

質疑応答

- オーバーヘッドはどう考えるか？
 - 基本的には考えなくても問題ない。
- チューニングツールはあるか？
 - JMeter というのがあるが、背景知識がないと使いこなせない。
- 性能と言っても実際のアプリケーションがないと、測定できないと思うが？
 - 実際のアプリケーションがない場合、類似システムの経験やある種的前提をつけて考えざるを得ない。特にうまいやり方がある訳ではない。
- 不特定多数の場合、どのようにシミュレーションするか？
 - ある種的前提をつけて考える方法をとる。
- GCが行われるまで使用済み変数を解放しないJavaは使わない方がよいのではないか？
 - 実際には仕事上使わざるを得ない。

5.初回参加者の自己紹介

平中幸雄氏、加藤亨氏、丹治雅行氏、中島理恵子氏より挨拶があった。

以上(記載者:滝口 亨)