



負荷テストの極意 第二回

株式会社アシスト
矢野英也

イントロダクション

皆さんこんにちは。今回は、連載第一回目ということもあり、連載させていただく経緯についてと負荷テストを大きく4つのフェーズに分けて、その各フェーズに関しての特徴を簡単に説明しました。第二回目となる今回は、負荷テストのフェーズでもテスト成功の明暗を分けると言っても過言ではない、「計画フェーズ」についてのお話です。

何事においても、計画は最初にするべき事項であると思います。計画がしっかりと立案されていないとその先のステップが大きく影響をうけることになるは皆さんも十分ご承知いただいていると思います。今回は、その重要な負荷テスト計画の中でも以下の3つのポイントに絞って解説していきたいと思います。

- ・ 負荷テストの目的と種類
- ・ 負荷テスト環境(構成要素)について
- ・ 負荷テストにツールは必要か

負荷テストの目的と種類

まず負荷テストの目的について考えます。通常負荷テストの目的を考える際に、担当者として陥りがちなのは、ユーザーが体感する応答時間や秒あたりのヒットなどを真っ先に考えてしまうことです。大局的に見て、「そのシステムがユーザーに対してどのようなサービスを提供すべきか。」といった観点から考え始めて目標の内容をブレイクダウンしていけば、本来の負荷テストの目的を見失わずにテストを完了させることができるでしょう。また、目的を考える上で、必要なことは負荷テストの種類を理解しておくことです。負荷テストの種類は、以下の表のように大きく分けて4つに分類できます。

テストの種類	概要	具体例
性能テスト	・業務プログラムの多重度と同等の取引を発生させ、約束した性能が出ているか検証 ・想定した負荷時のレスポンス状況やシステム状況を把握し、実運用上問題がないかを確認するために実施	50ユーザで参照系トランザクションと更新系トランザクションをミックスして実施する
限界テスト	・業務プログラムにおける多重度(関連するオブジェクト数)の倍以上の取引を発生させてシステムの限界を確認する ・システムの限界状況を把握し「レスポンスは遅くなるがダウンしない」などの状況を確認する	登録ユーザは50人の場合100ユーザでアクセスさせてシステム状況を監視
連続テスト	・ある一定の取引を長時間連続的に発生させ、その状況で運用しても問題ないか検証 ・ログやファイル領域はパンクしないか、メモリリークが発生していないかなどを確認	24時間連続稼働させた場合にどのような状況が発生するかを把握する
比較・再現テスト	・チューニング前後の性能比較、プログラム変更時のデグレードの確認、サーバごとの性能比較などを行う場合は同一条件下でテストを実施	性能問題が顕在化している場合に、原因を特定するためにテストを実施する

表 1. 負荷テストの種類

また、目標を設定する際は、ネットワーク、データベース、アプリケーションなどの各担当者と認識を合わせておくことも重要です。各担当者間での負荷テストの目的のずれがあったり、思惑の違いにより、せっかく実施したテストが無駄になるようなことが無いように注意が必要です。

負荷テスト環境(構成要素)について

次に負荷テスト環境について考えます。一言で負荷テスト環境といってもいろいろあるので以下のようにまとめてみました。

- ・ 対象のシステムにおけるアプリケーションサーバやスイッチ、ストレージなどのハードウェア
- ・ 対象のシステムにおける OS、データベースやアプリケーションなどのソフトウェア
- ・ 関連(もしくは連携)するシステムの要素

それぞれの要素をシステム構成図としてきちんとまとめておくことで、問題発生時やテストの条件を考慮する際に正しい視点で負荷テストを検討することが可能になります。

また、構成を考えるにあたり重要なポイントがもう一つあります。それは、負荷を構成要素のどのポイントから発生させるのかということです。なぜなら、負荷を発生させるポイントによっては、実施したいテストが実行できないことや取得したいデータが取れないということにもなりかねません。

また、本番環境とテスト環境でのハードウェアの性能(機種)の違いやソフトウェアの設定についても注意が必要です。私の経験では、ハードディスクの性能が本番環境に比べてテスト環境ではかなり劣っていたために、せっかく実施したテスト結果が正しく評価できなかった例もあります。せっかく実測でテストをしても本番環境とテスト環境の差を机上の計算で埋めてしまったことで、想定より性能を出せないためにハードウェアの追加の費用が発生することもありました。理想は、本番環境と同じか同等の環境で負荷テストを実施することがベストであることは疑う余地はありません。

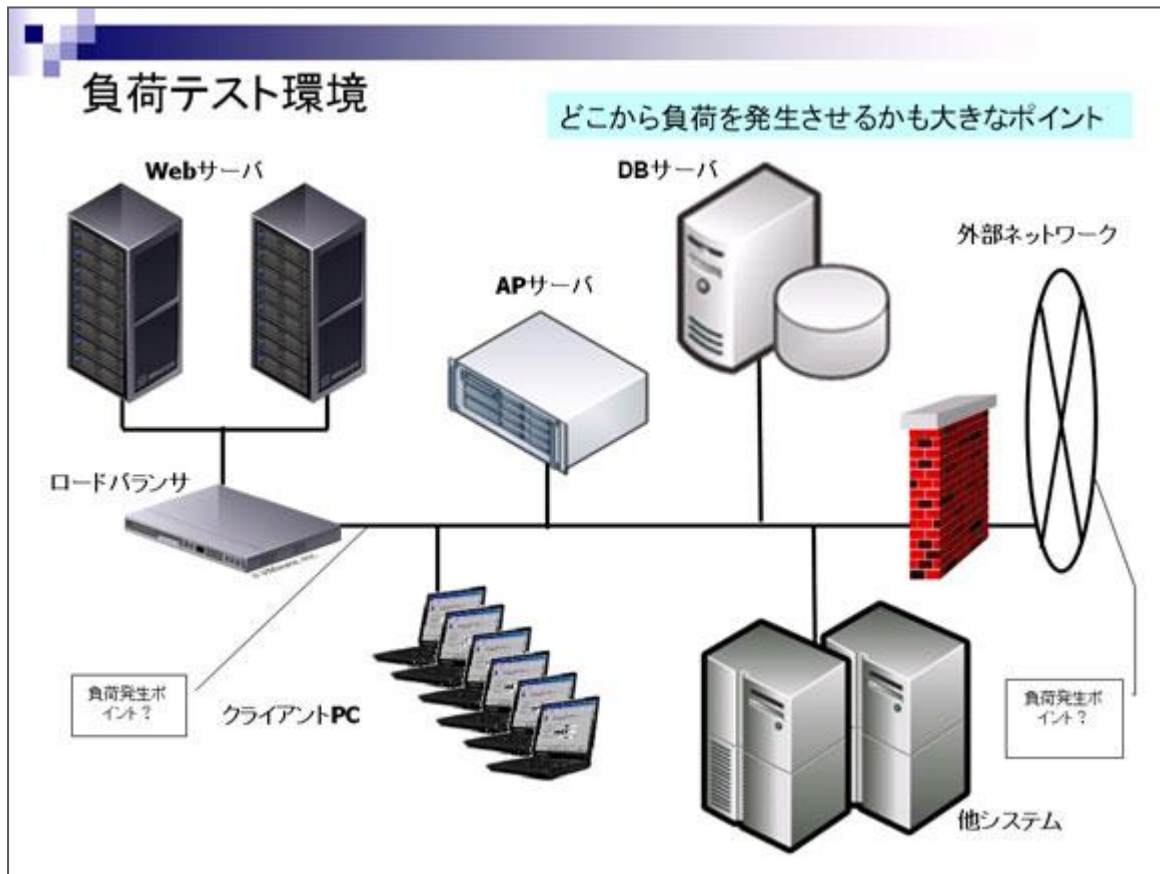


図 1. 負荷テスト環境の構成要素

負荷テストにツールは必要か

最後に、負荷テストにツールは必要なのかということに関して述べたいと思います。結論から言うと、有償・無償に関わらず何らかのツールは持っていたほうが良いと思います。理由としては、上記負荷テストの種類の項目で説明したように、いろいろな目的の負荷テストを実施するためには、一昔前のような手動で負荷テストをすることや負荷テスト用プログラムをその都度作成して実行するだけではカバーしきれない、もしくは正確に負荷テストが実施できないからです。

ツールを検討する際には以下のポイントで考えるといいと思います。

- ・ スクリプト作成、シナリオ作成の効率性
負荷を発生させるためには、仮想ユーザというユーザの代わりとなってシステムにアクセスさせるスクリプトが必要になります。その作成の効率性は、負荷テストツールに求められる重要な要件のひとつです。対象の環境によっては、記録できなかったり、スクリプトのチューニングが多く発生する場合もあり、事前の確認が重要です。
- ・ 分析ツールの効率性
負荷テストは、システム開発の最終フェーズで、残された時間が限られている場合がほとんどです。そのため、

簡単に早く負荷テストをできることだけではなく、レポート化することや簡単に分析することができるかも大きな要素です。

- ・ モニタリング項目
ツールで可能な収集項目が多いほど、ボトルネックの特定をしやすいなど多角的に分析しやすくなります。
- ・ マルチプロトコルへの対応
環境ごとにテストツールを変えるのは非効率です。どんな環境でも同じようなテスト準備やアウトプットが出力できると、計画の立案や緊急時の対応への適用可否など工数が見積もりやすくなります。
- ・ スケーラビリティ
最近の負荷テストはユーザ数の観点でも大規模になってきていることも多く、一台の端末から多くの負荷を発生させられるツールが望ましいです。

今回は、負荷テストの計画について述べてきましたが、詳細なドキュメントとして BTO Club の以下のサイトにドキュメントして掲載していますので参考にしてみてください。

クラブ – Tips ページ「成功する負荷テスト計画の立て方 12 のポイント」

尚、テスト計画に関してのコラムは、本 BTO Club では、日本ヒューレット・パカード株式会社湯本剛さんの「テストマネジメント虎の巻」の連載においても、私とは違った視点で触れられていますので、ご覧になってみてください。

次回は、今回の計画に基づいた「負荷テストの準備」について書く予定です。お楽しみに！