TDS Performance Improvements

Interim Release Report

Prepared for: Smarter Balanced

Fairway Technologies, Inc.
January 26, 2016

Executive Summary

Smarter Balanced and Fairway Technologies are pleased to release an update to the SmarterApp test delivery software that incorporates an initial set of performance improvements. Under contract with Smarter Balanced, Fairway will continue to optimize performance and add important new features. The final software release under this contract will occur Spring 2016 and will be ready for the 2016-2017 school year.

Performance tests use cloud servers hosted on Amazon Web Services. One key metric is the time it takes for a student to start a test from the moment they select their test until the first test question is presented. Under a modest load of 10,000 concurrent students the unenhanced software required 4.0 seconds to present the test. The enhancements reduce this time to 0.9 seconds.

Another important metric is the number of students that can concurrently login and take a test. According to Fairway's tests, while the unenhanced software could provide tests for up to 20,000 concurrent students, this new release provides tests for up to 35,000 concurrent students which is a 75% improvement in performance.

Performance Improvements

Overview

Fairway started its first round of performance improvements by running profilers and load tests to identify the main bottleneck of the current open source code. The Fairway team found that the TDS MySQL database interaction was the primary area of contention. By viewing the CPU usage, connections, writes/sec and reads/sec, Fairway determined the heaviest workload was produced during student login and the starting of a test.

After reviewing the performance metrics, the development team reviewed the code and profiled the API calls made during the login and start test processes to determine what areas of the code should be considered as opportunities for performance improvements. Fairway discovered an excessive number of database queries are made during these processes, resulting in an increased load on the database.

Improvements for this release have focused on reducing the number of database calls by:

1. Combining multiple queries into a single query
    a. Several methods query the database to get data that is then utilized in the next query. Where possible, these queries were combined using join operators in a single query.
2. Using domain objects to pass data between methods
    a. Many methods in the application do not accept enough data to accomplish their task. These methods rely on helper methods that query additional information from the database. Fairway eliminated these additional queries by passing domain objects with the required data.
3. Adding caching
    a. Fairway eliminated a significant number of queries for data that do not change often by adding caching. This includes system configuration and test metadata.
4. Replacing temporary tables with data structures
    a. A significant number of database queries were eliminated by replacing temporary tables with application data structures. Although the temporary tables were small each operation on a temporary table required another round trip to the database.

Fairway has developed a suite of regression tests to cover the features present in the test delivery system. The final regression testing suite will be delivered in Spring 2016. Fairway has utilized the regression tests, along with newly developed integration tests and a manual QA process, to ensure the code enhancements have not broken existing functionality or introduced new bugs. The code being delivered has passed all regression tests that are in place.

Results Summary

Fairway ran multiple load tests with the unenhanced open source code and with the enhanced version after completing the performance improvements listed above. The application was tested with 10,000 concurrent students (10K), 20,000 concurrent students (20K) and 35,000 concurrent students (35K).

General Test Data

	Unenhanced 10K	Enhanced 10K	Unenhanced 20K	Enhanced 20K	Enhanced 35K
Concurrent Students	10,000	10,000	20,000	20,000	35,000
% Proctors Start Session	100%	100%	100%	100%	100%
% Students Complete Test	100%	100%	98.6%	100.0%	99.9%

Database Info

	Unenhanced 10K	Enhanced 10K	Unenhanced 20K	Enhanced 20K	Enhanced 35K
Max database CPU	19%	8%	51%	17%	34%
Max writes per second	1,458	1,539	1,926	1,509	2,280
Max reads per second	173	125	159	129	272

Important HTTP Requests (time in milliseconds)

	Unenhanced 10K	Enhanced 10K	Unenhanced 20K	Enhanced 20K	Enhanced 35K
Start Test	2,191	471	5,682	2,808	5,395
Load TestShell.aspx	82	40	215	146	415
Fetch first 2 pages	1,709	340	3,383	1,915	5,365
Get Page Content	45	38	67	65	128
Total Test Start Time	**4,027**	**889**	**9,347**	**4,934**	**11,303**
% Change	-	**-78%**	-	**-47%**	-

Important Methods (time in milliseconds)

	Unenhanced 10K	Enhanced 10K	Unenhanced 20K	Enhanced 20K	Enhanced 35K
T_Login	658	22	886	144	401
T_OpenTestOpportunity	1,258	399	1,569	2,117	4,710
T_StartTestOpportunity	2,539	418	4,295	2,855	4,606
T_InsertItems	1,222	141	2,025	897	2,826

* All “stored procedure” times are included below for reference

Load Testing

Since Fairway was not provided the load testing plans used by AIR and Measured Progress, a new load test suite was developed using Apache JMeter. Without any knowledge of the previous plans used, Fairway cannot provide an "apples to apples" comparison against results obtained by AIR and/or Measured Progress. Instead, Fairway is providing a new performance baseline based on the unenhanced open source code retrieved from the BitBucket repositories on 12/14/2015.

For the initial performance improvements, Fairway's developers have been focusing on the number of students that can login and take a test with a ramp up time of 20 minutes. The final deliverable and requirement in the RFP allows the workload of those users to be spread out over time: 50% of the students login initially with the remaining 50% logging in 10 minutes later. Focused on getting as much performance impact as possible, the load test suite does not currently spread the load amongst two login groups. Fairway wanted to initially focus on the "worst case" scenario and fallback to spreading out the users later for the final numbers if appropriate.

The load test plan was created by recording all of the HTTP calls created when a proctor logs in and approves students, as well as the student logging in taking the test. The test includes all API calls to the server in addition to calls made from the browser requesting resources such as CSS files, javascript files, images, audio and video resources, etc. Since all of these resources are served from the TDS application it is important to send that load to the server to make the simulation as realistic as possible.

Criteria

From the original RFP, the Client Performance Requirements states "It should take less than 15 seconds to start a new testing session." There are multiple requests that contribute to this overall time. After the student clicks the final button to start the test, the following HTTP calls are made before the first question is loaded:

- Start Test (/student/Pages/API/MasterShell.axd/startTest)
- Load TestShell with resources (/student/Pages/TestShell.aspx)
- Fetch first 2 question meta-data (/student/Pages/API/TestShell.axd/updateResponses)
- Get page content (/student/Pages/API/TestShell.axd/getPageContent)

In calculating the unenhanced baseline for concurrent students and the final number of concurrent students reached, Fairway evaluated the combined averages of these calls and required them to be below 15 seconds in order to qualify.

Additional Details

AWS Environment

The following environment was configured for use across all load tests performed.

Component	AWS Instance	JVM Settings	Notes
ART DB	c4.2xlarge 8 vCPU, 15 GB RAM		
ART Web	c4.2xlarge (load balanced) 8 vCPU, 15 GB RAM	-XX:+UseConcMarkSweepGC\ -Xms2048m -Xmx10240m\ -XX:PermSize=1024m\ -XX:MaxPermSize=2048m\	
OpenAM	m4.xlarge (x1 load balanced) 4 vCPU, 16 GB RAM		
OpenDJ	C4.8xlarge 36 vCPU, 60 GB RAM		
Permissions DB	m3.medium 1 vCPU, 3.75 GB RAM		
Permissions Web	m3.xlarge 4 vCPU, 15 GB RAM	-XX:+UseConcMarkSweepGC\ -Xms1024m -Xmx10240m\ -XX:PermSize=512m\ -XX:MaxPermSize=2048m\	
ProgMan DB	m3.medium 1 vCPU, 3.75 GB RAM		
ProgMan Web	m3.medium 1 vCPU, 3.75 GB RAM	-XX:+UseConcMarkSweepGC\ -Xms512m -Xmx2048m\ -XX:PermSize=512m\ -XX:MaxPermSize=1512m\	
TDS DB	db.m4.4xlarge		RDS Instance with ~200K tests taken across 6 types of tests
TDS Web	c4.4xlarge (x3 load balanced) 16 vCPU, 30 GB RAM	-XX:+UseConcMarkSweepGC\ -Xms5120m -Xmx28672m\ -XX:PermSize=512m\ -XX:MaxPermSize=2048m\	
TSB	m3.medium 1 vCPU, 3.75 GB RAM		

Stored Procedure Times

All times displayed are the average time in milliseconds.

	Unenhanced 10K	Enhanced 10K	Unenhanced 20K	Enhanced 20K	Enhanced 35K
AA_GetItemgroup	137	8	318	473	479
AA_GetNextItemCandidates	437	50	721	641	1,222
IB_GetTestAccommodations	30	6	46	16	62
P_ApproveAccommodations	141	109	69	177	317
P_ApproveOpportunity	64	38	25	119	128
P_CreateSession	41	36	19	49	22
P_GetTestsForApproval	30	27	26	314	481
P_InsertSessionTest	15	10	7	17	6
P_PauseSession_SP	44	37	96	125	1,074
SetOpportunityStatus	253	118	233	661	801
T_GetEligibleTests	82	59	179	241	540
T_GetOpportunityItemsWithValidation	119	32	332	232	616
T_InsertItems	1,222	141	2,025	897	2,826
T_Login	658	22	886	144	401
T_OpenTestOpportunity	1,258	399	1,569	2,117	4,710
T_StartTestOpportunity	2,539	418	4,295	2,855	4,606
_CanOpenTestOpportunity	119	10	525	93	257
_GetInitialAbility_SP	236	9	559	110	226
_GetTesteeTestForms	129	33	307	206	523
_GetTesteeTestWindows	210	71	297	221	829
_InitializeOpportunity_SP	2,117	387	3,546	2,700	4,963
_InitializeTestSegments_SP	1,322	236	2,832	1,918	3,570
_OpenNewOpportunity	986	332	910	1,653	3,928
_T_ValidateTesteeLogin	53	2	150	24	61
TOTALS	**12,242**	**2,590**	**19,972**	**16,003**	**32,648**
% Change	-	**-79%**	-	**-20%**	-