

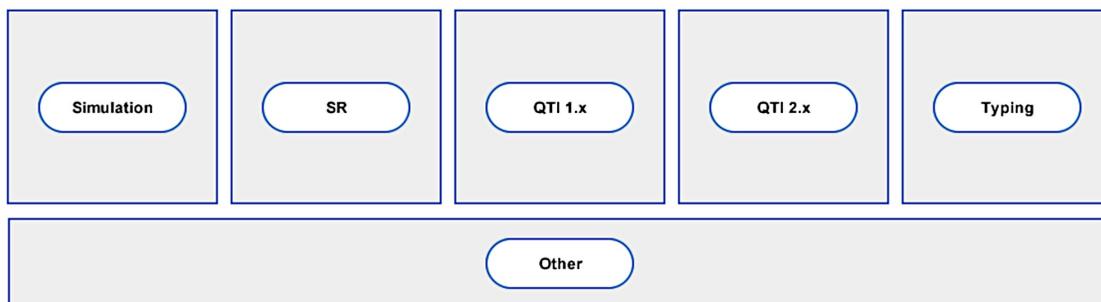
Hammer 2015

Platform Architecture

The Hammer framework was originally designed to render simulated environments that replicated the functionality to their real counterparts. In 2014, requirements for additional tests were built using Hammer (namely Selected Response and Typing). In 2015, additional renderers are going to be used within the Hammgmaier environment (namely a QTI engine supporting 1.x and 2.x items). Hammer is quickly evolving from a framework to a platform. Do to such requirements it is proposed to restructure Hammer to support a renderers and plugin architecture.

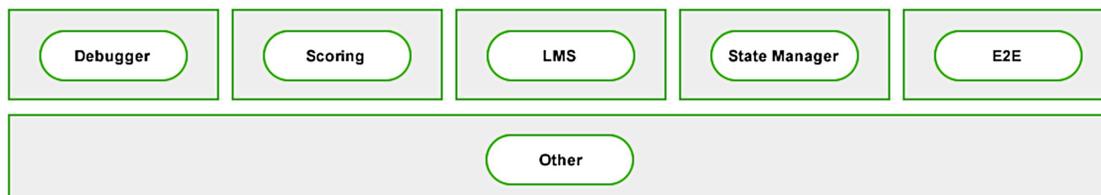
Renderers

Renderers manage components and other plugins required to display the question item. Renderers are responsible are composed AngularJS directives, which manage data and display the data according to the properties.



Plugins

Plugins are swappable libraries that typically perform a specific task(s), such as I/O, debugging, scoring, state management, etc. A question is composed of one or more plugins defined in the configuration of either the engine configuration file and/or the question configuration file.



Event Driven System

In order to keep the engine, renderers and plugins decoupled, an event bus will be used to pass information between components.

Backwards Compatibility with 1.1.x

With these changes, an attempt will be made to remain backwards compatible with version 1.1.x, and/or require limited changes to make the 1.1.x compatible with 1.2.x.

QTI 1.x & 2.x Support

Hammer currently has a custom Selected Response for handling Multiple Choice, Drag n Drop, & Hotspots. In order to better accommodate questions developed in PearsonVue's Exam Designer (ED), a QTI engine supporting 1.x and 2.x will be developed in Hammer as a renderer.

Editor & Debugger

An initial version an inline editor was developed for 1.1.x. Additional features will be added for better development support.

- Schema definitions to help
 - Auto-populate models
 - Validate data
- Debugger -- visual changes in order to make debugging easier to use (This will be broken out into a plugin)

Unit Tests

Currently less than 1% of the Hammer framework has any forms of unit tests. Unit tests are a fundamental part of any application they help:

- Reduce bugs in new features
- Reduce bugs in existing features
- Provides good documentation
- Improves design
- Constrains features

End-to-End (E2E) Plugin

Adding E2E will reduce the time it takes to tests builds against Hammer built exams and the framework itself. Currently, as changes are made, a human has to re-test questions and/or functionality to ensure the integrity of a question, exam and or framework. This E2E engine will allow a person to record and playback changes testing the integrity of the item. This will greatly benefit developers during the development process and will dramatically save time on all fronts. The same tests

can be shared between QA and developers to ensure changes and/or fixes made have either been corrected or have not violated and already working version.

Better Support for Drag n' Drop within Simulations

Drag n' drop support had been added for Selected Response items. Simulations could also benefit from this feature if globally supported.

Simulation widgets

In addition to the existing widgets that exist for simulations there are some new widgets that would prove beneficial such as:

- Menu
- Layout using Flexbox
- Polymer Styled Angular Widgets and Components
- Others TBD

Upgrading to Angular 1.3.x and upgrading other support libraries

Angular version 1.2.x has been deprecated in place of Angular 1.3.x. Upgrading the framework to the newest version will ensure continued support for future revisions of Angular. In addition to Angular, Hammer has other legacy libraries that are either no longer needed or should be upgraded. The upgrade will greatly reduce the complexity in the Hammer structure. Namely, removing the dependency of using *require.js* for compiling and accessing resources within the environment. The ability to compile will become more streamlined and accurate for revisions and builds. It will also reduce the files size.