

# Impact of Totara forking from Moodle

## Introduction

In September 2015 Totara Learning announced the decision to progressively fork Totara Learn from Moodle in order to allow us to make deeper architectural changes we believe are necessary to modernise the platform and provide key functionality required in the corporate learning technologies space. If you'd like to know more about this decision our CEO Richard Wyles published an article [Focusing on our mission – an open source "fork" of Moodle](#) that explores this in depth.

This document provides more details on the likely impact of the fork on organisations that wish to migrate from Moodle to Totara Learn, or are reliant on Moodle plugins or themes.

## Approach to divergence

Our overall approach to diverging from Moodle is that we will do so only out of necessity, when we believe making the required changes provide a better overall solution for our customers and provide a more robust platform for future developments. There is no benefit to introducing unnecessary differences so when possible we will continue to use Moodle's plugin architecture to extend rather than modify core code. When a core improvement is necessary we will make every attempt to introduce the change in a backward compatible way so existing code developed for Moodle will continue to function without requiring changes.

In addition, our decision to fork does not mean that we will immediately cease merging in code from the Moodle project. Initially we intend to continue to merge Moodle major versions and that process will continue for some time. Eventually we may reach the point where we stop merging directly, but we will likely still merge select features or bug fixes of relevance to Totara Learn. The impact of the fork will be dependent on which features are implemented in both Moodle and Totara Learn, and how those differences interact with each other. This in turn will determine how long we can support upgrade paths from Moodle and also plugin and theme compatibility.

## Known and unknown impacts

Assessing the future impact of the decision to fork can be difficult because some of the impacts depend on factors outside our control. The direction which Moodle chooses to take may significantly impact how quickly we diverge. There are however some areas that we are aware of or that we have more control over. Details on the expected impact of our fork on those areas is given below.

## Competency Frameworks and Learning Plans

Totara Learning introduced competency frameworks and learning plans in the very first release of Totara Learn. Recently Moodle have made the decision to add [Competency Based Education](#) to Moodle using the code from an earlier Totara Learn release as a starting point. This is currently it is targeted for inclusion in Moodle 3.1.

Since the same functionality will exist in both products merging 3.1 into Totara Learn is likely to lead to significant conflicts or duplication in functionality. Our handling of this will depend on the details of the Moodle implementation. If the Moodle implementation is largely unchanged, it should be possible to migrate competency and learning plan data from Moodle's model to Totara Learn's model and support a full upgrade from Moodle 3.1 to a later release of Totara Learn. If Moodle have made changes to the code to add functionality that we see value in we may be able to merge that functionality into Totara Learn. On the other hand, if Moodle have made complex data model changes or added functionality that is unsuitable for our customers it may be difficult to provide a smooth upgrade path. This will be assessed once Moodle 3.1 has been released and we begin the process of integrating parts of it into Totara Learn.

## UX / Themes

The lack of a clean modern look and the difficulty modifying the appearance themes has been an ongoing point of feedback for both Moodle and Totara Learn, and we feel that some fundamental changes are required in order to modernise the platform. While this work is taking a number of forms, there are two which are important in the context of compatibility.

### Templates

In order to give theme designers more control and flexibility we are prioritising converting Totara Learn to use templates. In order to minimise differences with Moodle we have decided to use mustache templates as they are already in use by Moodle. There are some areas where it might be necessary to extend Moodle's template solution to achieve what we need but we expect it to be backward compatible (existing Moodle code would continue to work unchanged).

### Theme changes

Moodle currently uses a modified version of the Bootstrap CSS framework in its "bootstrapbase" theme. The key difference is that the framework has been modified to apply Bootstrap appearances to Moodle CSS classes. Our preference is to modify the template HTML to use native bootstrap classes, therefore allowing Bootstrap to be used without modification. This may mean some HTML changes to core templates, which is likely to make Moodle and Totara Learn 2.2 - 2.9 themes incompatible with Totara Learn 9. While this work is being assessed we would like to be able to provide a "legacy" theme which retains the old HTML templates. This would allow earlier themes to continue to be used on Totara Learn 9 simply by switching the "parent" theme to use the legacy one. However the feasibility of this approach is still being assessed at this stage.

### Font icons

In Totara Learn 9 we are converting icons to use the "Font awesome" icon library. This will improve the consistency and appearance of icons in all themes. We will retain backward compatibility by allowing icons without a specified font icon to continue to use the original image version, and will make it straightforward to convert graphical icons into font based ones in third party code.

## Forms

We have decided to implement our own forms library to solve some of the limitations and problems with the current Moodle forms library. The new library will sit alongside the existing library so any forms based on Moodle's formlib will continue to function unchanged. The new library will provide significant new capabilities, but will be largely compatible with Moodle's library when building a form. Therefore converting a Moodle form to use the new library will be a very straightforward process in the majority of cases. Totara Learning will undertake the work to convert all the forms in Totara Learn extensions to the new library. The new library will likely offer some new capabilities not currently available in the Moodle forms library.

## Multi-tenancy

Multi-tenancy is a much requested feature, and one which is more important in the corporate space than an academic setting. Moodle did consider adding multi-tenancy to Moodle but the project was [cancelled](#) at the specification stage. We would like to add multi-tenancy into Totara Learn, but given the extent of the changes required it would be very difficult to implement in Totara Learn while still tracking the Moodle project across the board. Since there is no appetite for implementing multi-tenancy in Moodle this is one of the key reasons why we decided to fork.

Realistically implementing multi-tenancy is also likely to be one of the larger factors that will lead to divergence between Moodle and Totara Learn post-fork. Unless we can come up with a clever way to implement it without impacting existing Moodle code (unlikely) fundamental changes will be required that will make merging Moodle code significantly harder over time.

## Upgrade paths

The table below details the known and expected upgrade path support from versions of Moodle to Totara Learn. Upgrade paths from any earlier version of Totara Learn to a later version are always supported. If you have a client that is currently running Moodle and is considering upgrading to Totara Learn the safest course of action is to not upgrade beyond Moodle 3.2 since an upgrade path to Totara Learn will then definitely be supported.

Moodle version to upgrade	Minimum Totara version	Supported?
2.2	2.2	Yes
2.3, 2.4	2.4	Yes
2.5	2.5	Yes
2.6	2.6	Yes
2.7	2.7	Yes
2.8, 2.9	2.9	Yes
3.0	9	Yes
3.1	10	Yes, though the migration of Moodle competency/learning plan is not implemented
3.2	10	Yes
3.3	12	Yes

## Moodle plugins

Moodle plugins have never been officially supported in Totara Learn, but due to the similarity between the two platforms in practice most Moodle plugins function in Totara Learn without any problems. Since we intend to continue to support the existing API in a backward compatible way, we do not anticipate that this will change in the short to medium term. In the longer term it is possible that Moodle will introduce functionality that we don't include in Totara Learn, and plugins will be written that rely on that functionality, which will make those plugins unsuitable for use in Totara Learn.

If this becomes a problem we may consider building our own independent plugin repository and invest resources in testing, validating and updating Moodle plugins to ensure they work consistently in Totara Learn too. This effort would focus on the most commonly used plugins by corporate users and could be a community driven effort with assistance from Totara Partners or even subscribers.

## Themes

As mentioned above, significant changes to how themes are handled in Totara Learn 9 are likely to require new themes to be developed for version 9 and above. We will be attempting to provide backward compatibility via a "legacy" parent theme but at this stage it is unknown as to how well this will work.