# Experience API and OpenLabyrinth

OpenLabyrinth has now implemented the ADL Experience API (aka TinCan API or xAPI) which is designed to track activity data.

This work is based on Medbiq Learning Experience Working Group ([www.medbiq.org/learning\_experience](http://www.medbiq.org/learning_experience) )

## What is it for?

The [Experience API (xAPI)](https://www.adlnet.gov/adl-research/performance-tracking-analysis/experience-api/) is much extensively explained elsewhere but basically this is about activity streams and tracking what people actually do within our systems. We want to know, not just did they solve the case, but how did they get there, what resources did they use etc.

A lot of this has been tracked within OpenLabyrinth previously – this has been one of its strengths. But now with xAPI, we can track activities across multiple platforms and simulation systems. All of these activities are captured in a Learning Records Store (LRS).

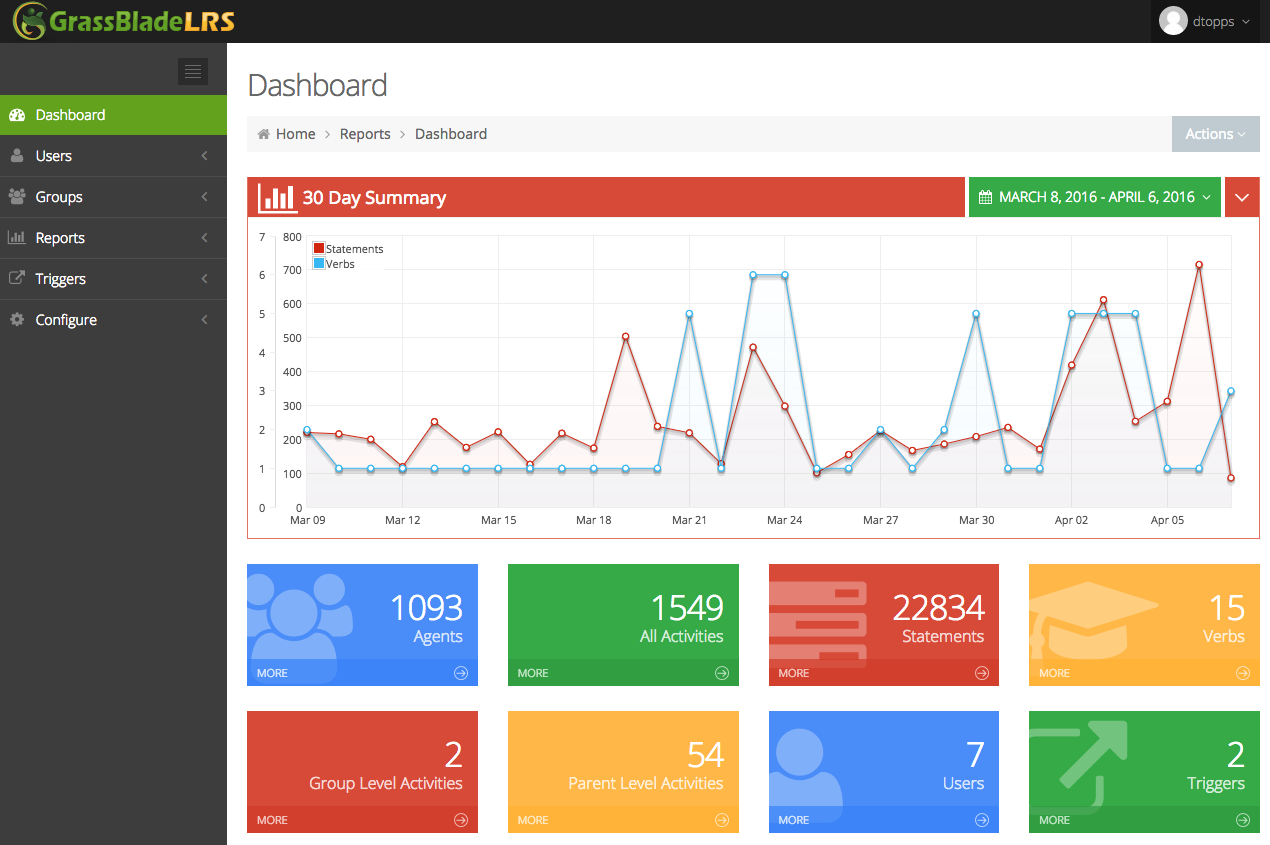
## Is this really Big Data?

We are using big data principles in how we store, analyze and visualize these activity streams. But this is not really big data. Even if we were to track absolutely every action by every learner, we would still be orders of magnitude less in data quantities than is produced by genomics, protein folding or the Large Hadron Collider.

## Can I track my own data?

Yes, you can. At the end of most cases, you can pull up an OpenLabyrinth Session Report which will show you most of this stuff. If you have the right privileges on OpenLabyrinth, you may also be able to do this for other cases and users.

You can now pull more detailed data from our GrassBlade LRS at <http://openlabyrinth.ca/grassblade-lrs> - if you need a login, just send us a request and we will consider it.



You can also set up your own personal LRS or request that data be forwarded to another LRS. There is some small processing cost to this and we may not be able to accommodate all requests but we are open to trying.

## How do I use xAPI reporting?

There are two ways in which you can use xAPI statements with your OpenLabyrinth cases. Most users will be happy with [*post-hoc* reporting](http://openlabyrinth.ca/setting-up-post-hoc-xapi-in-olab/). With this approach, you can generate statements on pretty much any set of cases or learners over the past 4 years on our demo server here. Yes, you can go way back before we implemented xAPI – pretty neat, huh? This is because our OpenLabyrinth servers internally tracked much of this data previously. It is comparatively simple now to export this to a LRS.

For some cases, you may want [real-time reporting of xAPI statements](http://openlabyrinth.ca/setting-up-realtime-xapi-reports-in-olab/). If done to excess, this could slow our servers down for everyone else so please be responsible. You also have to modify your case design somewhat to get useful real-time results.

## I am interested in doing more

If you have a serious interest in exploring activity streams from a research perspective and would like to collaborate on this, please contact us. We have a number of projects on the go just now and are always happy to consider further such collaborations. [Contact us through this link](http://openlabyrinth.ca/support/help-for-developers/).

For medical education, MedBiquitous has been particularly helpful. If you are interested in learning more about the Medbiq Learning Experience Working Group, check their web site. Always looking for new active participants.

## What is a Learning Record Store (LRS)?

An LRS is a different kind of database, specifically designed to store xAPI activity statements. For a much more extensive explanation, see here. They are designed to efficiently handle a very large number of statements. There are many ways of then analyzing the data. They can accept activity streams from multiple sources at once in a seamless secure manner. We are using an LRS to track xAPI data from OpenLabyrinth, WordPress, our H5P widgets, and several other simulation devices.

We have tested OpenLabyrinth with the following LRS database engines:

* [GrassBlade LRS](http://www.nextsoftwaresolutions.com/) – cheap and simple to maintain
* [SCORM Cloud](http://scorm.com/scorm-solved/scorm-cloud-features/) – very standards-compliant
* [Wax LRS](http://www.saltbox.com/) – stricter statement handling; more analytics
* [Watershed LRS](https://www.watershedlrs.com/) – seriously powerful with great analytics
* [Learning Locker](https://learninglocker.net/) – open-source, based on MongoDB
* [SCORM Engine](http://scorm.com/scorm-solved/scorm-engine/) – simplest way to bridge xAPI to SCORM

There are many others out there and the list is growing weekly. If you are interested in getting OpenLabyrinth to work with a particular LRS, [feel free to contact us](http://openlabyrinth.ca/support/help-for-developers/). For information on [how to connect OLab to an LRS, see here](http://openlabyrinth.ca/connecting-an-lrs-to-olab/).