

ALEX AMIES, ECAI WORKSHOP, MARCH
2019

**BUILDING BLOCKS FOR OPEN ECOSYSTEMS
SERVING ONLINE BUDDHIST COMMUNITIES**

AGENDA

- ▶ Technologies from the Fourth Industrial Revolution that relate to information systems serving Buddhist communities
- ▶ Characteristics of an open online ecosystems
- ▶ What are the building blocks that can push Buddhist information systems forward?
- ▶ Experience from NTI Reader and HB Reader

TECHNOLOGIES BEHIND THE FOURTH INDUSTRIAL REVOLUTION

- ▶ The Fourth Industrial Revolution
 - ▶ Schwab explains that the most revolutionary impact is not just the new technologies themselves but it is the amplification of the interconnectivity between these technologies (Schwab 2017, pp. 1-3).
- ▶ What are these technologies and how are they interconnected?
 - ▶ Development - open source software with source code management systems like Git that allow structured collaboration from multiple developers
 - ▶ Development - module systems based on open source software that enable (a) automation and (b) breaking down monolithic systems into components
 - ▶ User experience - mobile and web clients supported by cloud services

CHARACTERISTICS OF OPEN ONLINE SYSTEMS

- ▶ End user access. Examples:
 - ▶ Open: Linkable sets of web pages
 - ▶ Closed: Proprietary systems where the links end at a login screen are closed. Example: university libraries where only students can login.
- ▶ Re-use of content. Examples:
 - ▶ Open: Licenses that allow re-use of content (eg, Creative Commons). Parallel with lending of physical books.
 - ▶ Closed: licenses that restrict re-use are closed. Example: publishing restrictions preventing authors from making their works publicly available.
- ▶ Software
 - ▶ Open: Software ecosystems that enable collaboration across organizational boundaries (open source licenses)
 - ▶ Closed: Proprietary software that cannot be used across organizational boundaries (commercial licenses)

INNOVATION

▶ Collaboration

- ▶ Schwab: “when firms share resources through collaborative innovation, significant value can be created for both parties as well as for the economies in which such collaborations take place” (Schwab 2017, p. 60).

▶ Open Source

- ▶ GitHub currently reports 31 million developers with 96 million repositories, now owned by Microsoft (GitHub site)
- ▶ TensorFlow is a deep learning (artificial neural network) library released to open source by Google in 2015 and in 2016 became the most popular machine learning project on GitHub (Dean 2017).
- ▶ ***All*** the significant innovation today is happening in open source

OPEN SOURCE – CONFUSING MESSAGE

- ▶ Fairy tale - we all adopt open source and live happily ever after
- ▶ Reality
 - ▶ Not everybody wants to release their source code and digital artifacts to open source - this needs to be respected
 - ▶ Open source can be disruptive
 - ▶ Open source is harder because removing underlying dependencies requires work
- ▶ Refined message
 - ▶ Open source and private source can exist together
 - ▶ Open source does not require giving up copyright
 - ▶ Open source is more sustainable than private source
 - ▶ The value in collaboration is worth the effort in open sourcing assets

COLLABORATION

- ▶ Git - source code management system - created by Linus Torvalds, also known as the creator of Linux
- ▶ Enables multiple authors to contribute to code repositories
- ▶ Operations
 - ▶ **Add** or update files
 - ▶ **Clone** a repository
 - ▶ **Commit** changes
 - ▶ **Push** from a local repository to a remote repository
 - ▶ **Fork** a repository
 - ▶ **Pull** request (for users not authorized to commit directly)
 - ▶ **Merge**
 - ▶ Many more operations - create branch, rebase, etc.
- ▶ Enables code reviews, test automation on every commit with customization

MODULAR SYSTEMS

▶ Monolithic systems

- ▶ A single system has a pyramid of requirements usually representing a single stakeholder, eg business owner
- ▶ Not scalable - only a small number of developers know how it works
- ▶ Automation is very difficult - manual testing, long release cycles
- ▶ Developers not engaged (anonymous and lacking authority)

▶ Modular systems

- ▶ Requirements broken down by modules serving a diverse community
- ▶ Scalable - developers mainly need to know their own components and the external surface of other components
- ▶ Automation: automated testing, fast release cycles

OPEN SOURCE MODULAR ECOSYSTEMS

Examples

- ▶ Debian package management system (Debian Linux)
- ▶ Docker containers
- ▶ JavaScript (npmjs)
- ▶ Python Package Index (pip)
- ▶ Go lang package system (go get)
- ▶ REST web services
- ▶ gRPC (Google Remote Procedure Call)

EXPERIENCE WITH NTI READER AND HB READER

- ▶ NTI Reader was open source from the beginning (2013)
- ▶ HB Reader is not open source
- ▶ Makes extensive use of open source
 - ▶ Go lang, Docker containers, Material Design, MySQL
- ▶ Has some modular breakdown but not extensive
- ▶ Been lacking in active engagement from wider community

ENGAGEMENT

- ▶ Modularization is critical for engagement
 - ▶ Specific requirements can be met by a combination of more general components
 - ▶ Potential consuming projects have diverse requirements and
 - ▶ Challenges
 - ▶ Defining the components
 - ▶ Different technology choices by other projects
- ▶ Forums
 - ▶ Just recently added announcement groups (ntireader-announce and hbreader-announce)
 - ▶ No user forum or online group yet
 - ▶ No engagement with groups outside of Buddhist circles yet

MODULAR SYSTEMS USED FOR TEXT ARTIFACTS

- ▶ Different corpora
 - ▶ Base software at GitHub project [chinesenotes.com](https://github.com/chinesenotes.com) can adapt to different text corpora
 - ▶ Does not use a standard corpus format
- ▶ Dictionary
 - ▶ Multiple dictionaries can be used if formatted correctly
 - ▶ Cannot plug-in external sources yet
 - ▶ Does not use the standard TEI format

MODULE SYSTEMS FOR SOFTWARE

- ▶ Go lang
 - ▶ Two subsystems
 - ▶ Analyze corpus and build site
 - ▶ Web application
 - ▶ Cannot be retrieved using go get
- ▶ Python
 - ▶ Motley collection of scripts for vocabulary ingestion

USER INTERFACE

- ▶ JavaScript
 - ▶ Started modularization with the chinesedict-js module
- ▶ Authentication
 - ▶ Translation portal requires a private login
 - ▶ Initially developed proprietary system
 - ▶ Considering switching to Google authentication but need multiple methods

RESOURCES

▶ NTI Reader

- ▶ Website: <http://ntireader.org>
- ▶ Source: <https://github.com/alexamies/buddhist-dictionary>
- ▶ Announce group: <https://groups.google.com/forum/#!forum/ntireader-announce>

▶ HB Reader

- ▶ Website: hbreader.org
- ▶ Announce group: <https://groups.google.com/forum/#!forum/hbreader-announce>

▶ Chinese Notes

- ▶ Website: chinesenotes.com
- ▶ Source: <https://github.com/alexamies/chinesenotes.com>
- ▶ Announce group: <https://groups.google.com/forum/#!forum/chinesenotes-announce>

▶ Chinese Dictionary JavaScript Module

- ▶ Source: <https://github.com/alexamies/chinesedict-js>
- ▶ JavaScript Module: <https://www.npmjs.com/package/@alexamies/chinesedict-js>

▶ A Primer in Chinese Buddhist Writings

- ▶ Website: <http://www.primerbuddhism.org/>

REFERENCES

- ▶ Dean, Jeff. 2017. "The Google Brain Team – Looking Back on 2016." Google AI Blog. January 12, 2017. <http://ai.googleblog.com/2017/01/the-google-brain-team-looking-back-on.html>
- ▶ GitHub, 'State of the Octoverse,' <https://blog.github.com/2018-10-16-state-of-the-octoverse/>
- ▶ Loeliger, Jon and Matthew McCullough, 2012. *Version Control with Git: Powerful Tools and Techniques for Collaborative Software Development*. 2nd ed. O'Reilly Media, Inc.
- ▶ Microsoft. 2018. "Microsoft Completes GitHub Acquisition." The Official Microsoft Blog. October 26, 2018. <https://blogs.microsoft.com/blog/2018/10/26/microsoft-completes-github-acquisition/>
- ▶ Schwab, Klaus 2017, *The Fourth Industrial Revolution*, New York: Crown Publishing Group.