Expanding the Anki Envelope

Anki, the uncontested gold standard in assuring long-term retention of learnt material with Spaced Repetition, is far beyond being just another application. Anki is an Open Source ecosystem with hundreds of community-contributed addons, thousands of shared decks and a vibrant world wide community of users and developers. It is also a shining example of how an initiator and prime mover of an Open Source project can reap admirable financial gains from a niche in that ecosystem.

Model

There are innumerous educational applications, many of which incorporate a Spaced Repetition component, usually of secondary importance to the main thrust of the application. A central characteristic shared by nearly all of them is what we might term *“import only”* meaning that they can import material from various sources (including Anki decks) but enable exporting only in their own limited proprietary format or limit their exporting in accepted formats such as Anki to only certain characteristics but not all. We propose turning the model on its head by placing Anki’s universal Spaced Repetition at the core and enabling developers to produce custom applications around that core and spawn profitable commercial initiatives.

The following is based on the rich experience I have acquired as an EFL English teacher over past years in the use of Anki in a school setting.

Anki in Education, the challenge

Anki in institutional education is fundamentally different from the individual user model around which Anki is built. Furthermore, the educational model described here can be seen as a template, applicable in many other multi-user environments.

Individual learners who leverage Anki to accelerate and retain their learning are motivated, disciplined self-learners maintaining a daily practice routine. They invest time and effort into creating new cards and organizing their decks. But for few exceptions, schoolchildren of all ages exhibit none of those qualities.

Neither are overloaded teachers willing to invest time in learning Anki or even so much as to learn about Anki at all. Anki in a class environment demands tedious, repetitive maintenace actions while many commercial educational software products offer educators markedly better ease-of-use.

There is a dearth of attractive Anki decks suited to the class-specific curriculum. Finding and sifting existing sources demands effort.

Expanding Anki's use beyond the limited scope of independent learners into institutionalized education **would broaden the ecosystem hundreds of times over** and open myriad commercial opportunities.

Limitations today's single-user paradigm, partial list

1. Requires installation and user responsibility for decks
2. Steep learning curve, interface virtually unintelligible to school pupils
3. No group-wise maintenance and statistics capabilities
4. Lack of friendly gamified interfaces
5. No commercial incentive to content (deck) producers
6. Many, many more serious limitations…

Examples

1. Generate a list of cards in the 11th grade chemistry deck which reached maturity over the past month from those students with over 15 practice sessions during the month.

2. Generate a list of students who practiced Anki less than three times a week over the past month.
3. Switch the current practice deck for all of my 5th graders.
4. Correct a card in the 7th grade history deck.
5. Enable certain addon capabilities for all subscribers.

Note: Numbers 4 & 5 require no distribution of the corrected deck or addon.

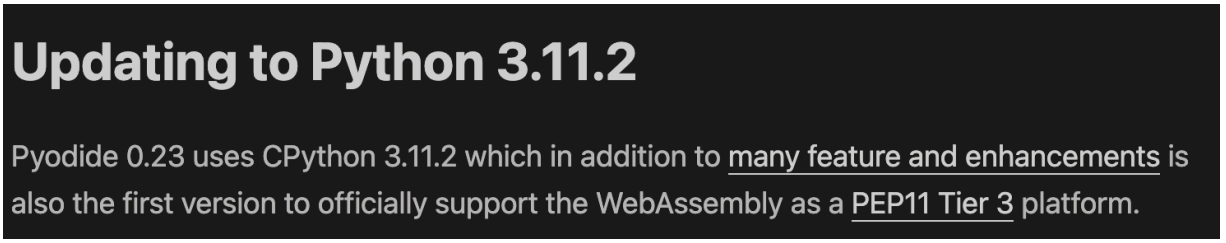
The contrived examples should suffice to show how trivial mass maintenance chores and statistics accumulation should be. They need not be performed by the learner at all but rather by the teacher/administrator. In fact, maintenance, statistics and perhaps card creation interfaces should be in separate applications altogether, certainly not there to confound the learner.

Proposed Direction: Web-based Anki with multi-user DB

Until recently such an undertaking would have been a formidable challenge necessarily entailing a clean break from the Anki codebase. **The recent development of Wasm becoming a web standard is a game changer**, making compiling existing code to Wasm a viable point of departure.

Suggested Stages

1. Get Anki running in the browser by compiling to Wasm.
   As of 31.3.2023 Pyodide now officially supports Wasm:

   PySide (vs. Pyodide) is the preferable version for compiling Python for QT applications to Wasm but lags behind Pyodide in Wasm support. Anki's GUI is in QT primarily because Anki is a desktop app. We see no need to maintain QT compatibility for the web version. In any case, the

web browser capability built into Anki becomes superfluous when running in a browser.

2. Replace Anki's Sqlite database with a multiuser DB. See SeaQL, especially SeaORM including SQLite migration tools.
3. Disengage user scheduling and statistics from Anki decks to enable subscribing to decks without requiring distribution.
4. Enable DRM (or similar) content protection of decks and content. This is essential to spawn a commercial marketplace for content (such as proprietary decks) providers.
5. Spawn the marketplace for (4) above as commonplace in many Open Source projects such as WordPress.
6. Incorporate provisions to enable hosting providers to offer custom UIs or (what today are called) add-on capabilities.
7. **The crowning jewel:** AI-based content generator to automate production of high quality, curriculum-aligned decks with many more sophisticated features than presently available. This might be a closed source project made available to content (deck) producers on a subscription basis.

Strategic Challenges, (at least) 5 broad avenues

1. Educational strategies

These are pedagogical concerns, having little to do with Anki per se but are essential to achieving universal adoption of Anki or Spaced Repetition as an educational tool.

2. Paradigm shift from Single-User to Class model

a. Repeated installation of Anki with the same addons and decks is utterly superfluous. There is no need to push the same identical decks into all Anki installations in the class or school. Neither do the students have to browse or edit their own cards. Class-wide statistics should be available to the teacher or administrator at the flick of a switch. These are but a few of the essential characteristics of the Class-wide Anki model.

b. The obvious solution is Web-based Anki with a central database in the cloud. Until now, this implied a complete rewrite of Anki from the bottom up. Tools such as PyScript, Qt for WebAssembly, Rust and WebAssembly and SQL.js indicate that Anki as a web application could be realized with few modifications to the code.

3. Publishing high quality decks

While Anki is Open Source, there is ample room for opening space to launch commercial ventures within the Anki ecosystem, common in other Open Source ecosystems, to fill the void in which educational oriented decks are few and far between, not aligned with school curricula or textbooks, definitely not on a par with the quality of those found in commercial educational products. This becomes doubly important in school environments both in order to achieve academic goals and maintain student interest.

4. Commercial hosting and Gamified front ends

   Following the previous item, here too offering a horizon for commercial initiatives in hosting and quality gamified front-ends would greatly advance wide scale adoption of Anki in educational settings.

5. Interim Solutions

   Anki can be adopted in school settings before a fully webified edition is launched. There is a real need for addons which can offload the burden of repetitive operations, providing ease of operation for whole classes on a level comparable to commercial offerings.

Implement a daily practice regimen

Move Anki practice out of the classroom and into a daily curriculum time slot A new subject which we'll figuratively call ***Retention***, during which decks from multiple subjects will be practiced. While this might sound counterintuitive, it is precisely the way Anki is used by independent learners. Take medical school for example

Advantages:

1. Relieves the teacher of having to learn and administer Anki.
2. Saves valuable teaching hours.
   Anki can be implemented and monitored by non-teaching staff.
3. No need to "*sell*" Anki to teachers who have neither time nor interest in it. One key figure in the school is sufficient.
4. Provides teachers with highly focused per-individual-student learning statistics vs. today's fuzzy overall picture.

Note that the new subject, ***Retention***, requires no additional instruction hours. To the contrary. Its class time comes from shifting inefficient, in-class, teacher-based review hours to the ***Retention*** classroom.

Produce high quality Anki Decks suited to curriculum

This is a formidable challenge, one which makes commercial offerings attractive. As Open Source, currently not focused on institutionalized

education, this is relegated by default to the community which will not on its own provide the necessary material without offering incentives.

To this end we propose a two pronged effort consisting of:

1. Establishing a portal along the lines of Shared Anki decks but much more user friendly to facilitate and encourage educators to produce and share **and sell** decks or use them on a subscription basis.
2. Launch a commercial initiative aimed at producing quality, copyright curriculum-based decks.