



Ecosystem infrastructure for smart and personalised inclusion
and PROSPERITY for ALL stakeholders

D204.2 Subtitle editor modification to better support real-time collaboration for near-real time captioning

Project Acronym	Prosperity4All
Grant Agreement number	FP7-610510
Deliverable number	D204.2
Work package number	WP204
Work package title	Media and Material Automated/Crowdsourced Transformation Infrastructures
Authors	Dean Jansen, PCF
Status	Final
Dissemination Level	Public/Consortium

Delivery Date **22/9/2016**
Number of Pages **18**

Keyword List

Collaboration, real-time, and user presence

Version History

Revision	Date	Author	Organisation	Description
1	07/08/2016	Dean Jansen	PCF	Initial Draft
2	08/08/2016	Martin Deinhofer	FHTW	Internal Review: Figure references were missing; image alternative texts exist; type: a user instead of a user! Contents: Missed some context information about the current system and some technical diagrams and detailed descriptions.
3	10/08/2016	Marco Aimar	ENG	Clarify technical relationships and diagrams, add technical detail to technical implementation of three column layout and language schema changes.
4	14/08/2016	Gregg Vanderheiden	RTF	Extend logic in first paragraph of executive summary to remainder of section. Reflect impact of tool improvements in broader P4A ecosystem.
5	1/9/2016	Dean Janssen	PCF	Incorporated Feedback

Table of Contents

- Executive Summary 6**
- 1 Contribution to the Global Architecture 7**
- 2 Introduction 9**
- 3 UE Subtitle Editor, History/Diff, & Data Model Enhancements..... 10**
 - 3.1 Initial Situation 10**
 - 3.2 Goals & Targets 11**
 - 3.2.1 "Write Lock" Collision Avoidance System 11
 - 3.2.2 Notes and Timecode Linking System..... 11
 - 3.2.3 New Data Model..... 12
 - 3.2.4 Editor Support for Dynamic Language Timing, Switching, and Comparison..... 12
 - 3.2.5 More Distinguished User Presence in History/Diff System 13
 - 3.3 Implementation..... 14**
 - 3.3.1 "Write Lock" Collision Avoidance System 14
 - 3.3.2 Notes and Timecode Linking System..... 14
 - 3.3.3 New Data Model..... 15
 - 3.3.4 Editor Support for Dynamic Language Timing, Switching, and Comparison..... 17
 - 3.3.5 More Distinguished User Presence in History/Diff System 19
- 4 Conclusion..... 20**
- Annex I: Glossary 21**

List of Figures

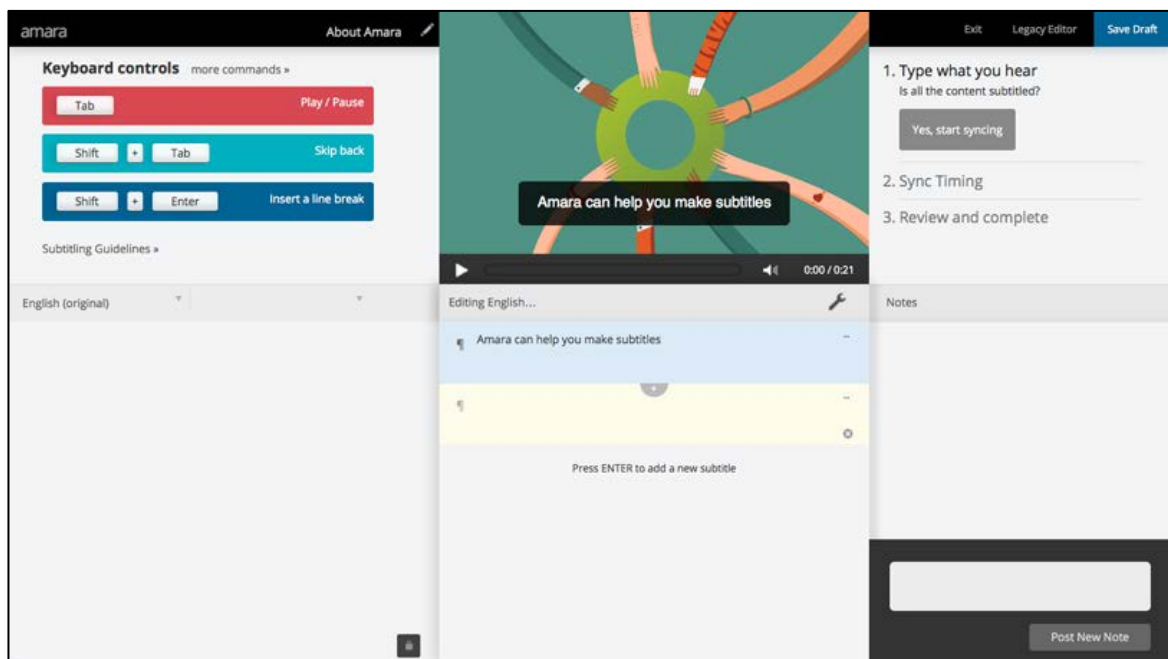
- Figure 1: UE caption & subtitle editor 4
- Figure 2: Overall picture of Prosperity4all, with WP204 highlighted 5
- Figure 3: The UE’s original, vertically stacked, translation UI..... 11
- Figure 4: The write-lock dialog 12
- Figure 5: The notes section in the subtitle editor, with timecode hyperlink..... 13
- Figure 6: Diagram of original data model 13
- Figure 7: Diagram of new data model..... 14
- Figure 8: Three-column layout for subtitle editor 15
- Figure 9: Dynamically switching languages in the subtitle editor..... 16

Executive Summary

The Participatory Culture Foundation (PCF) hosts and develops Amara, a platform for crowd/community created captions and translated subtitles. Amara is undergirded by the Unisubs Engine (UE), an open source platform that will be included as an infrastructural building block in the P4A's DeveloperSpace.

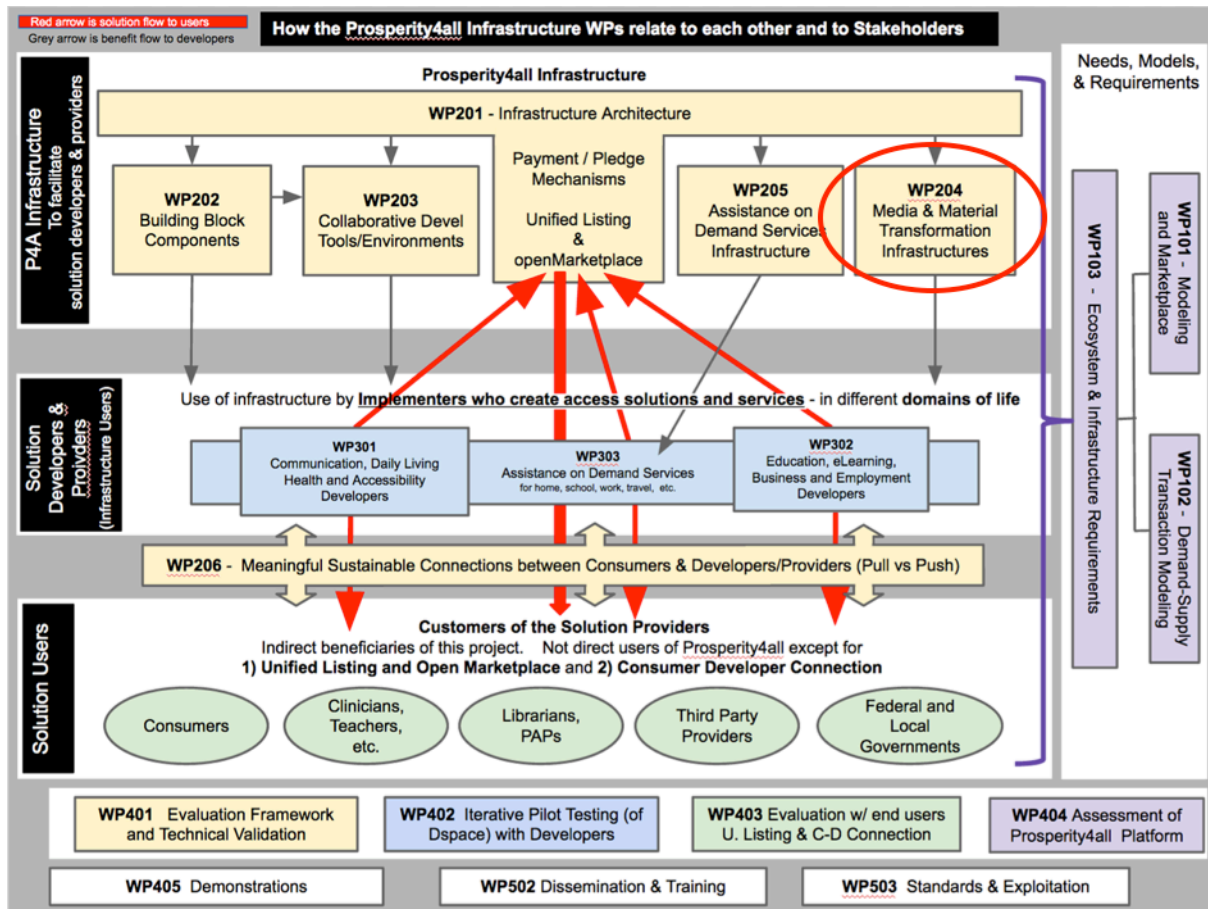
This document provides a summary of improvements made to the UE that will facilitate better collaboration between people who are captioning and translating videos, with the ultimate goal of making captioning easier and less costly. D204.2 accomplishes this goal by enhancing the UE subtitle editor (see fig. 1) for better real-time collaboration, enabling people to caption and subtitle in communities.

Figure 1: UE Caption & Subtitle Editor



1 Contribution to the Global Architecture

Figure 2: Overall Picture of Prosperity4all, with WP204 Highlighted



The UE is an infrastructure module available in the P4A Infrastructure; it provides an infrastructure or platform that can be used by both vendors and communities to caption and/or translate videos for greater accessibility. The source code and infrastructure for the UE will be available through P4A's DeveloperSpace.

Prior to being part of the P4A consortium, PCF launched Amara as a prototype, which evolved into an award-winning platform for community-driven captions and subtitles. Although the underlying software, the UE, was open source since the beginning, it was not particularly modular or easy to extend.

The goal, in improving the UE as a member of the P4A consortium, is to increase the platform's flexibility, modularity, compatibility, and ease-of-use – ultimately driving broader and more cost-effective accessibility for video and the ability for other developers to add

other enhancements (see fig. 2). The improvements to the UE are being made as improvements to the code and documentation (as in D204.1), as well as additional features. In addition, the P4A-driven features are adding compatibility with additional video services, enable users on mobile devices to participate in community accessibility efforts, and enable community members to work together in real time to caption and subtitle video.

2 Introduction

The purpose of the T204.2 enhancements in WP204 are to make the UE –which is the open source software and platform that undergirds Amara– more powerful for any developer or organization that would like to engage a community for creating captions or subtitles, with the ultimate goal of making captioning easier and less costly. To accomplish this task, the team focused on a variety of key improvements to the UE that make it more modular, make it easier to extend, and will better support real-time and near-real-time collaboration on captions and subtitles.

The T204.2 caption enhancements, are primarily focused on the UE subtitle editor, and include: notes and timecode linking system to facilitate better real-time collaboration; a "write lock" to avoid collisions when a user is working on captions; a new data model that enables dynamic many-to-many relationships between subtitle languages; dynamic switching between languages; and tools to support comparing two differently timed subtitle languages side-by-side, while in edit mode. With additional features to support greater user presence coming in a future set of refinements for T204.2.

3 UE Subtitle Editor, History/Diff, & Data Model Enhancements

As noted above, the caption feature enhancements for T204.2 are clustered around the UE subtitle editor. The three subsections in this section review the initial state of the subtitle editor, the goals and targets for the enhancements, and, finally, the execution and implementation of the enhancements.

3.1 Initial Situation

The UE began life as a prototype interface for captioning and subtitling web video. The prototype evolved into a more robust caption and subtitle editor that won awards for advancements in accessibility and multi-culturalism.

The original goal with the UE was to create a "Collaborative Wikipedia-like-system for captioning and subtitling;" however, with the original subtitle editor, coordinating work on captions with more than one person at a time was still challenging. This was an obstacle to real-time collaboration, where two people would work together more seamlessly on captions or translations.

There were two basic issues that made real-time work difficult. The first was accidental overwriting of captions or subtitles – if two people were editing simultaneously, and one finished sooner, the second person's edits would overwrite the first. For real-time collaboration, this was a showstopper. The second issue was that users could leave comments about captions for another user to see, but these comments appeared outside of the editor and were not in the context of the actual work being done. Alleviating these issues would make real-time collaboration and review of captions and subtitles significantly easier.

The UE, while useful for translating captions into different languages, lacked an important ability that could transform it into an incredibly powerful system for web based subtitle translation: it did not support timing changes between different language translations. In other words, if you were translating from Japanese to German, you were stuck with the exact timing and number of subtitle lines as the Japanese original. German phrases are often much longer than their Japanese counterparts, which can lead to a sub-optimal experience for the viewer, with German subtitles appearing rapid-fire. To have a truly robust system, the UE would need to support each language being timed independently. Additionally, for a true

many-to-many system, the ability to change which language was being displayed and translated from, in the editor, would also be important.

Additional opportunities were identified for creating a more robust and engaging experience for collaborating on captions and translations via the presentation of a more distinguished user presence, especially in the caption history/diff comparison system. The history/diff comparison system is where a user can see who has done which portion of edits and contributions to captions or translations. The addition of greater user presence would enable a user to not only see who is making the best contributions to a current project, but might also be a good metric for finding collaborators for future projects.

3.2 Goals & Targets

Each of the major areas for improvement in the previous section will be broken down into greater detail in this section. The implementation section will review the final output and realization of these features in the UE.

3.2.1 "Write Lock" Collision Avoidance System

A primary challenge with the initial subtitle editor was the possibility for "edit collisions" – this would occur when two users were editing the same set of captions simultaneously. The first user to save would have their work overwritten once the second user saved their work. The most straightforward solution to this issue would be implementing a "write lock" system, where only one user would be able to edit at a time – if a second user attempted to edit, they would get a message noting that the captions were in-progress. This functionality unlocks possibilities in terms of real-time collaboration, where one user creates a partial or full draft and a second user reviews and improves the work, without fear of overwriting the work of a collaborator.

3.2.2 Notes and Timecode Linking System

While powerful for a single user, the editor lacked functionality that would enable two people to easily collaborate in real-time, checking and improving each other's work. A system for adding notes, embedded in the context of the subtitle editing experience itself, enables faster and more direct collaboration. In addition to adding commentary in context the notes could, optionally, be linked to specific lines of captions. The ability to "hyperlink" a note to a specific caption line would enable a user to seek to the context of a comment

rapidly. This type of specificity in collaborative feedback fosters greater low-friction real-time collaboration between two or more users.

3.2.3 New Data Model

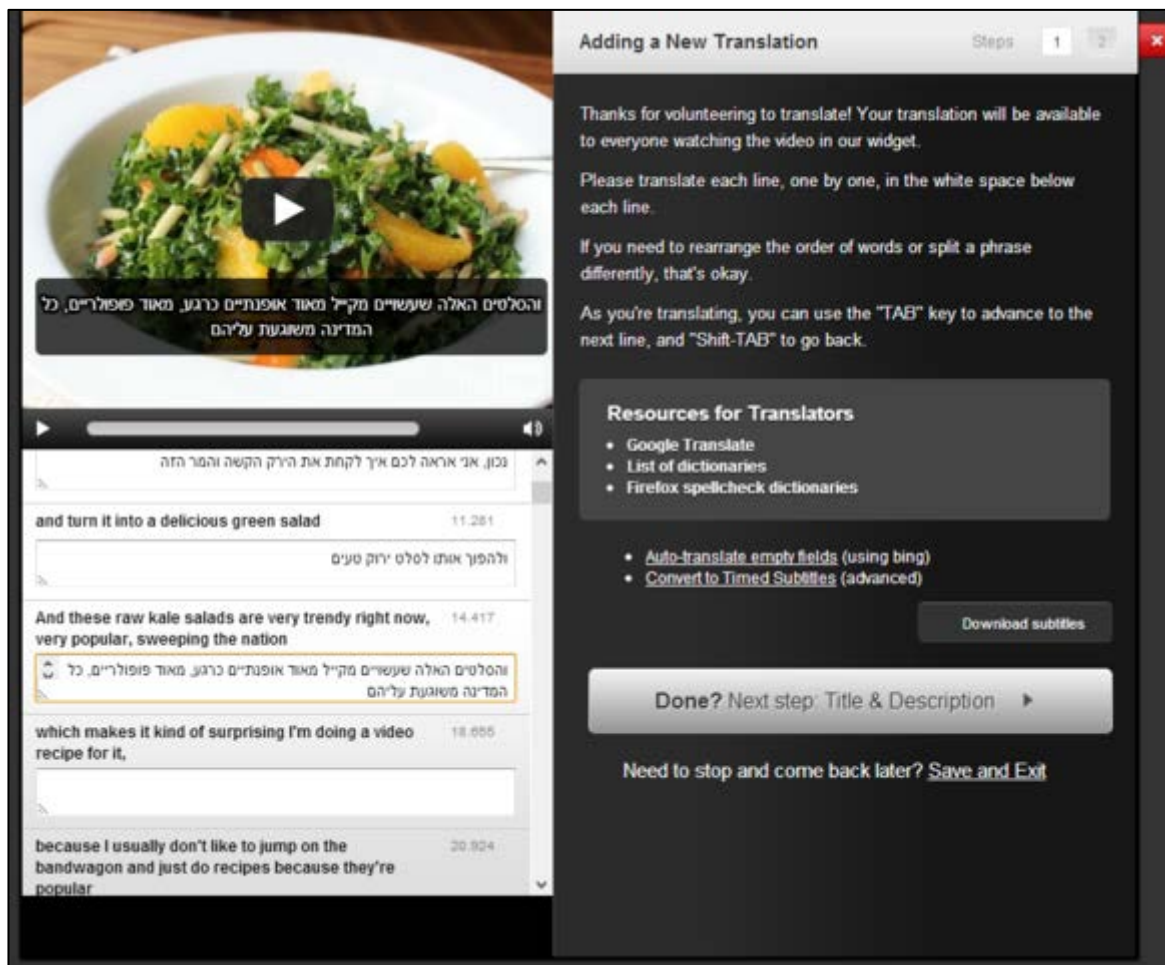
The underlying data model for the UE relied on an "initial language caption", for a video, which would set a fixed timing/line template that subsequent translations would use. In other words, every translated subtitle had to exactly match the timing and number of lines as the original language caption. While technically simple, and easy to accommodate in a basic subtitle editing User Interface (UI), this system was brittle and frustrating for users doing translations. For instance, when one language had longer translated lines, the space and time allotted for subtitles could be much too short (or vice versa). This led to user frustration and sometimes caused people to drop translation projects.

To overcome these shortfalls, a new data model was developed, where each language would have its own dynamic timing and number of lines. This improvement would alleviate many of the problems above, and create a flexible and vastly more powerful system. Additionally, the subtitle editor UI would require improvements to take advantage of the more powerful underlying data model – the necessary improvements are described in the next sub-section.

3.2.4 Editor Support for Dynamic Language Timing, Switching, and Comparison

The original UE subtitle editor had a separate interface for captioning and translating. For subtitle translation, it used a common and relatively straightforward vertically stacked system. The editor displayed the "original language" –that the user was working from– and the "target language" –that the user was translating into– stacked on top of one another (see fig. 3). In order to create a more dynamic relationship between the two languages, the UE team redesigned the editor UI, with input from communities of captioners, and placed the original language in a column adjacent to where the user was translating into their target language. To add even greater flexibility, the column with the original language could be dynamically toggled to show any currently available language.

Figure 3: The UE's original, vertically stacked, translation UI



3.2.5 More Distinguished User Presence in History/Diff System

The history/diff system allows any user to review (and optionally compare) every set of edits in the history of a set of subtitles. Greater user presence in this context has the potential to foster deeper connections between contributors in a community. A more distinct user representation, via avatars, can signal which users are most active in a more memorable way

than plain text. The original implementation of the history/diff system did not include much in the way of user presence, beyond simple text notation for a user's edits. The improved history/diff system will give a richer and more memorable feel, including user's avatars, for which user(s) edited areas of a caption or subtitle.

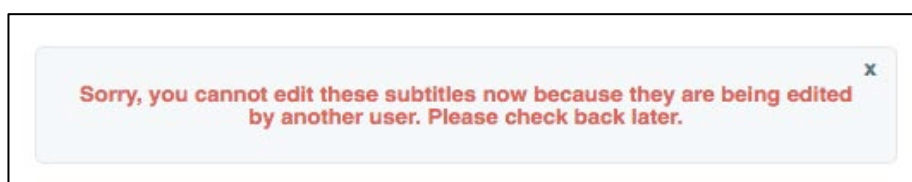
3.3 Implementation

Each of the major areas for improvement in the previous section will be broken down into greater detail in this section. The implementation section reviews the final output and realization of these features in the UE.

3.3.1 "Write Lock" Collision Avoidance System

The "Write Lock" system is relatively straightforward. On the back-end, a system was developed that locks sets of subtitle (i.e. languages) that are currently in-progress. This ensures that a user cannot accidentally overwrite the work of a fellow collaborator, who is working at the same time. On the front-end, a simple UI was developed to alert any user that tries to edit subtitles that are currently in-progress. If a user is inactive, the lock will automatically save work and time-out, allowing others to edit the subtitles (see fig. 4).

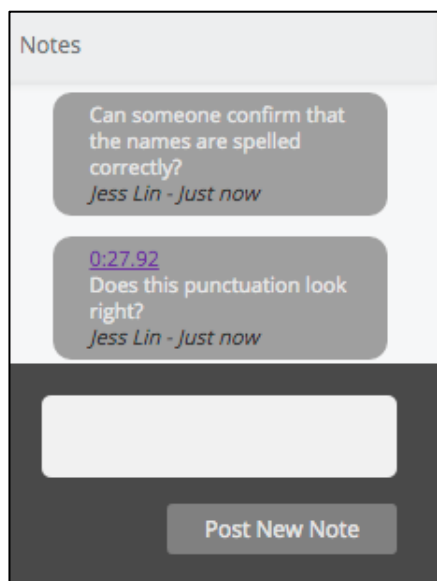
Figure 4: The write lock dialog



3.3.2 Notes and Timecode Linking System

The notation system is integrated with the UE subtitle editor UI, and utilizes the right-hand column for reviewing and adding notes. Any user who is actively editing subtitles in the editor can review any pre-existing notes, as well as create new ones that other users can see (see fig. 5). Another option, for a user editing subtitles, is to leave a timecode-linked note, which will allow anyone reviewing the note to easily "jump" to the timing and specific context of the note. A timecode-linked note can be created via the context-menu, which is available on each subtitle; this way a user can include a link to the exact point in the subtitles directly in their note. It's a powerful way to encourage rapid real-time engagement.

Figure 5: The notes section in the subtitle editor, with timecode hyperlink



3.3.3 New Data Model

The new data model is, from the end-user perspective, a big change in how captions and translated subtitles are related to one another inside UE. Instead of locking the timing of translations to the original "parent" captions (see fig. 6), the languages have a many-to-many relationship that allows more fluid and precise translation (see fig. 7). The translated subtitles can optionally "piggyback" on the original caption's timing –this can speed the process of translation up– but they are not compelled to maintain this original timing in a context where it doesn't make sense.

Figure 6: Diagram of original data model

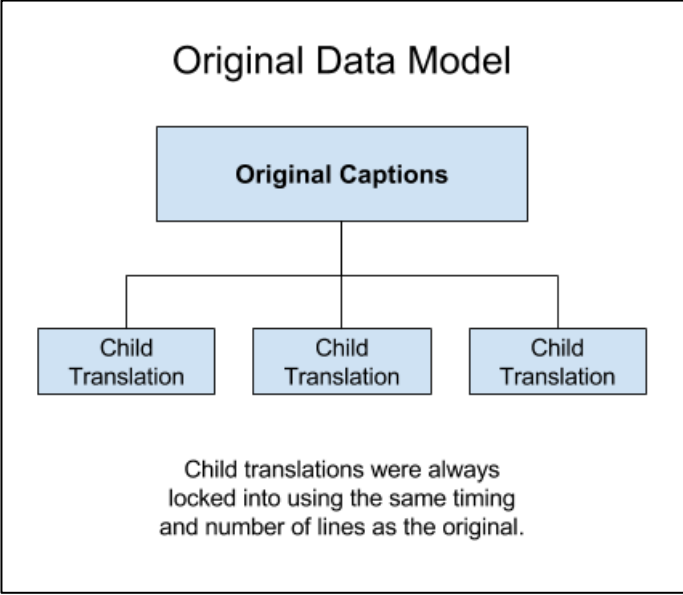
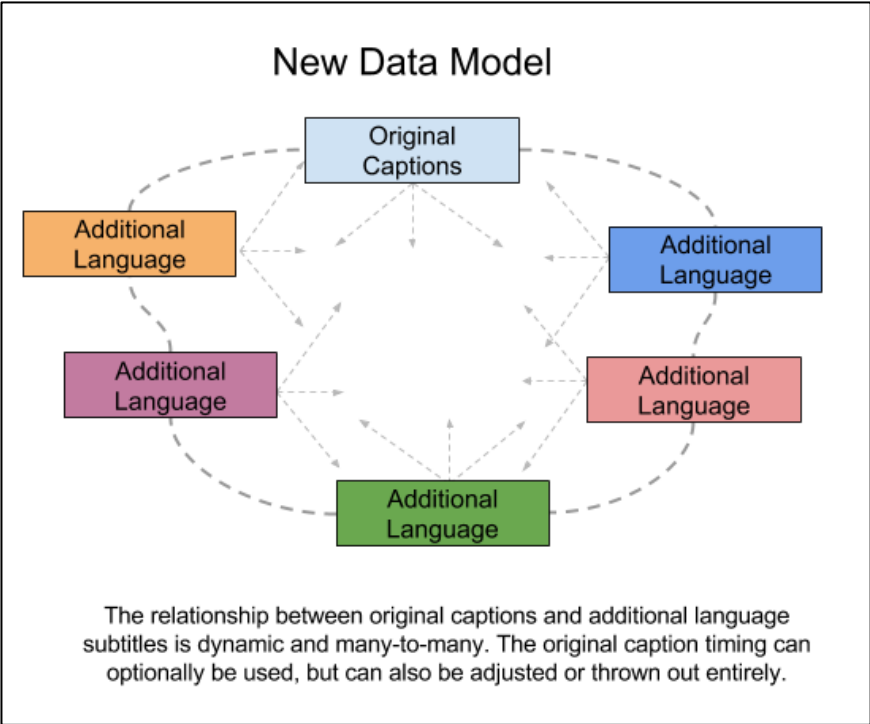


Figure 7: Diagram of new data model



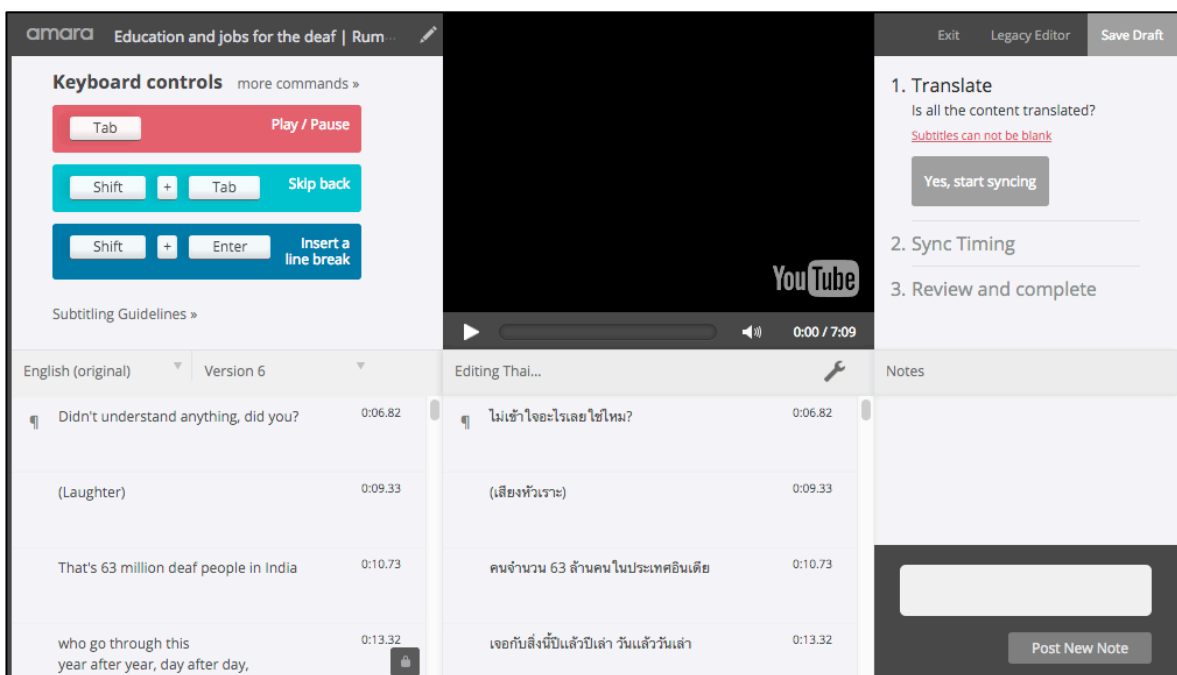
Materially, many of the changes are reflected in the structure of the UE's database and caption/subtitle management code. Initially, child captions drew on the database entry for the parent caption's timing – the child captions could not be timed independently. In the new database schema, the timing of child subtitles is uncoupled from parent captions.

Practically speaking, the user is exposed to the new data model through the UE subtitle editor – the user experience, which has been greatly enhanced, is described in the next section.

3.3.4 Editor Support for Dynamic Language Timing, Switching, and Comparison

In order to confer an enhanced user experience and greater flexibility, the new data model necessitated some changes to the UE subtitle editor. The most drastic change was moving from a two-column vertically stacked translation layout to a three-column layout. The horizontal orientation enables the source and target languages to be decoupled, which means there can now be more (or fewer) subtitle lines in one language than in another translation.

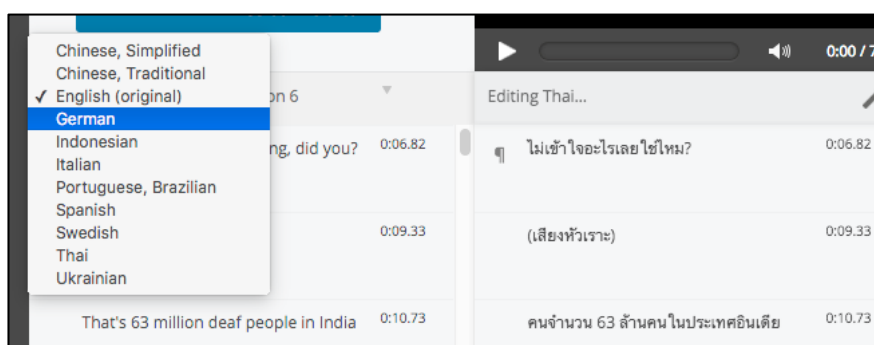
Figure 8: Three-column layout for subtitle editor



Another major difference is that translated languages can have differently timed subtitles than the original captions. This is especially useful in cases where a language has longer or shorter translations – the timing for subtitles appearing on screen can be perfectly tailored to each language, for any given video.

Another unique feature in the three-column layout is the ability to dynamically switch between source languages. This was simpler to enable, due to the additional space created by a three-column layout. It allows a user to choose which version of the source language they would like to translate from, in case they would prefer to use a bridge language. For example, imagine a video with Tagalog dialogue – it is initially captioned into Tagalog; a person who speaks Tagalog and Spanish could translate the video into Spanish; at a later time, another person –who speaks French and Spanish– could use the Spanish translation as the basis for translating into French, even if they didn't understand the native Tagalog captions.

Figure 9: Dynamically switching languages in the subtitle editor



The ability to bridge and piggyback off various languages in order to create dynamically timed subtitles that are perfectly tailored to each language, opens up amazing possibilities for cross-cultural connection through video. Currently, UE is the only web-based subtitle platform that enables this level of flexibility when dealing with caption translation.

3.3.5 More Distinguished User Presence in History/Diff System

As stated in the DOW, this feature will be presented in a future refinement of D204.2.

4 Conclusion

Much work has been done to improve the UE modularity, and extensibility and to enhance the subtitle editor's performance for real-time collaboration. The modularity improvements can enhance the ability of developers to incorporate or extend the functionality of the UE. The T204.2 feature improvements can benefit any person or organization that wishes to create more cost-effective captions or subtitles with the help of a community. This approach goes to the heart of UE's, Amara's, and Prosperity4All/GPII's vision for a more collaborative world that fosters broader participation and engagement.

Source code for the Unisubs Engine: <https://github.com/pculture/unisubs/>.

The Unisubs Engine Editor can be seen working in a production environment at: <https://amara.org/en/videos/create/> (note: videos/captions added to the Amara.org become publicly viewable/editable).

Annex I: Glossary

Abbreviation	Full form
AAL	Ambient Assisted Living
ACS	AsteRICS Configuration Suite
AoD	Assistance on Demand
API	Application Program Interface
AsteRICS	Assistive Technology Rapid Integration & Construction Set
AT	Assistive Technology
C4A	Cloud4All
D	Deliverable
DoW	Description of Work
DSpace	DeveloperSpace
GUI	Graphical User Interface
GPII	Global Public Inclusive Infrastructure
ICT	Information and Communications Technology
IDE	Integrated Development Environment
ISO	International Organization for Standardization
IT	Information Technology
KPI	Key Performance Indicator
P4A	Prosperity4all
PCF	Participatory Culture Foundation
R&D	Research and Development
RtD	Read the Docs
REST	Representational State Transfer
SP	Sub-Project
UE	Unisubs Engine
UI	User Interface
UX	User Experience

Abbreviation	Full form
VM	Virtual Machine
WP	Work Package