

MAKING ON-LINE SCIENCE COURSE MATERIALS EASILY TRANSLATABLE AND ACCESSIBLE WORLDWIDE: CHALLENGES AND SOLUTIONS

Wendy. K. Adams, *Department of Physics, University of Colorado, Boulder, Colorado 80309, USA*

Hisham Alhadlaq, *The Excellence Center of Science and Mathematics Education, King Saud University, and the Department of Physics and Astronomy, College of Sciences, King Saud University, PO BOX 2455 Riyadh, Saudi Arabia*

Christopher V. Malley, *PixelZoom, Inc., Boulder, Colorado. USA*

Katherine K. Perkins and Jonathan Olson *Department of Physics, University of Colorado, Boulder, Colorado 80309, USA*

Fahad Alshaya and Saleh Alabdulkareem, *The Excellence Center of Science and Mathematics Education, King Saud University, and the Department of Curriculum and Instruction, College of Education, King Saud University, PO BOX 2458 Riyadh, Saudi Arabia*

Carl E. Wieman, *Department of Physics, University of Colorado, Boulder, Colorado 80309, USA and the Carl Wieman Science Education Initiative, University of British Columbia, Vancouver, British Columbia, V6T 1Z3, Canada*

Abstract

The PhET Interactive Simulations Project recently partnered with the Excellence Center of Science and Mathematics Education at King Saud University with the joint goal of making simulations available worldwide. One of the main challenges of this partnership is to make PhET simulations easily translatable so that truly anyone with a computer can use them in their classroom. The PhET project team has created the Translation Utility that allows a person, who is fluent in both English and another language, to easily translate any of the PhET simulations. This can be done with minimal computer expertise, making the translation process accessible to faculty and teachers. In this presentation we will share solutions to many of the unexpected problems we encountered that would apply in general to on-line scientific course materials including working with a language that is written right-to-left, different character sets, possible misconceptions and various conventions for expressing equations, variables, units and scientific notation.

1. Introduction

As a part of the PhET Interactive Simulations Project we created a method for translating our science and math simulations (sims) and then hosting them on our website. This approach works because the underlying science is universal and it's only the words and norms used to represent it that must be translated. In this paper we present our current translation process and discuss the pitfalls we encountered along the way. By documenting the process, we hope others can avoid many of the difficulties we encountered while creating such a tool.

The PhET Interactive Simulations Project is a substantial (~85) and growing suite of professional quality sims for teaching and learning science. The sims are written in Java or Flash and are distributed from the PhET website <http://PhET.colorado.edu> at no cost to users, with roughly 10 million uses in the past year. The majority of PhET sims are for teaching physics but there are a growing number in chemistry, biology, math and other sciences. PhET sims provide a high degree of interactivity in terms of user control, dynamic feedback, and multiple representations. The sims enable students to make connections between real life phenomena and the underlying

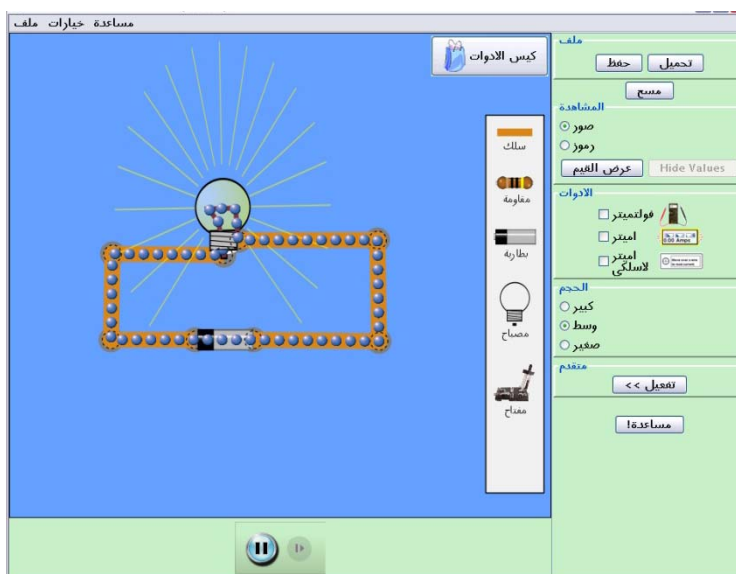


Figure 1 shows a screen shot of Circuit Construction Kit translated into Arabic.

science which explains such phenomena (Fig. 1). Our team of scientists, software engineers and science educators use a research based approach – incorporating findings from prior research and our own testing – to create sims that support student engagement with and understanding of scientific concepts. Our research includes investigating the use of PhET sims in a variety of educational settings. (PhET, 2009).

In this paper we will describe our current solution for translating the sims and our website including a few brief examples of the technical concerns involved. We also explain the method we settled on for finding translators and our co-authors from the Excellence Center of Science and Mathematics Education (ECSME) at King Saud University will share their experience as users of our Translation Utility. Creating this method for translating sims was a difficult and time consuming process; however it was much more efficient than rewriting the sims from scratch. It also allows a single location to host the same product in multiple languages, giving the instructor the opportunity to choose which language s/he will use in their classroom.

2. Current Solution

Developing software for world-wide use involves addressing two related issues: internationalization and localization. Internationalization is the process of designing software so that it can be adapted to various locales without engineering changes (Deutsch, 2001). Internationalization is often abbreviated as i18n (there are 18 characters between 'i' and 'n'). Localization is the process of adapting internationalized software for a specific locale (language and country) (Deutsch, 2001). This paper will address one aspect of localization: translation. For technical details on how PhET has addressed other aspects of i18n and localization, see (Malley, 2009).

In December of 2007 we released the initial version of the Translation Utility (PhET Team, 2007) for our sims. The Translation Utility is a program that allows the translator to see the English strings as they enter their translation. The translator can then test their work in the actual sim to make sure it appears as intended. Once complete, the translator emails PhET the translated strings, we compile a translated version of the sim and make it available on the PhET website. This solution came after several other avenues for acquiring translations proved unsuccessful. In addition, it has recently become obvious that we also need to have the delivery mechanism, the website, in the user's preferred language.

2.1 Translators

The most successful method for translating sims turned out to be a contributor model similar to the Wikipedia editing model¹. Instructors who use PhET sims volunteer their time and expertise to create translated sims. These instructors are ideally suited for creating the translated versions. On the surface this contributor model appears to be the least expensive; but, there are much more fundamental reasons. Bi-lingual teachers who use the sims have three necessary characteristics: They know the scientific terms and notation in both their language and English; They are using the sim in class so are familiar with the content, and; they have a personal interest in the final product because they need to use it in their course(s). We have had one other group that meets these requirements who were not actual instructors. A teacher in Romania created a class project where his students each translated a sim and entered it in a contest. These turned out to be of very high quality since the students were in the unusual situation of knowing the sims, knowing the science of both languages and having motivation to see the finished product. At the time of writing this paper the Translation Utility had been available for 20 months and we had 1054 translations of sims.

It is useful to review our failed models. We first tried to use Google Translate to translate the website and sims into Spanish which often resulted in incomprehensible, or highly inappropriate, Spanish phrasings. Next we tried hiring bilingual students, but it turned out to be very difficult to find students with the necessary science background. Neither of these approaches led to a high-quality product and neither had the potential to lead to translations in 40 languages.

One of the challenges for the translator includes keeping the string length to a minimum. PhET sims use very minimal text which means there is less to translate but it can be a challenge to express the idea in minimal wording. In addition, sometimes there is not a direct translation so it is

¹ Wikipedia's editing model is not to be confused with their translation model which has fully independent entries.

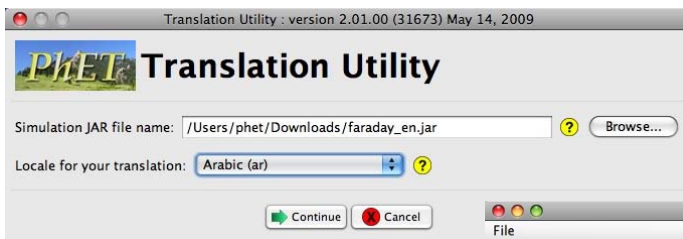


Figure 2 Shows the two windows of the Translation Utility. A. The upper right depicts the initial window where the translator chooses their locale. B. The lower right window shows the working window where the translator enters their translated text into the text boxes on the right side of the screen.



necessary that the translator completely understand the material and context of its use, which is one reason the translator usually has to be a teacher.

2.2 Simulation Translation Utility

It may appear that the contributor model is a cost efficient option; however, professional software developers are quite expensive, it took them 180 hours to create the translation utility. Much of this time was figuring out the most efficient design. Here we will describe much of what we learned through this process in an attempt to save another group time during the development stage. In addition to the actual utility, each sim has to be “internationalized” by creating string variables (Malley, 2009).

Translation of both Java and Flash sims is supported by the Translation Utility (PhET Team, 2007). The Utility provides a translator (a person creating a locale-specific translation) with a GUI (Graphical User Interface) for creating PhET string files. Translators choose their locale from a list of locales (Fig. 2A). The GUI then displays two columns: a left column shows the English strings, and a right column allows translators to enter the corresponding strings for their locale (Fig. 2B). A translation can be tested at any time by pressing a “Test” button, which will run the sim in the specified locale, with the translator’s strings. The translator should be checking that their translation is appropriate, is rendered properly, and does not create user-interface layout problems. Layout problems, can occur when a translated string is significantly longer or shorter than the original English string. When the translation is completed, pressing a “Submit” button creates a string file, which can then be emailed to PhET.

Early attempts to specify locale used only a language code. Feedback from translators quickly told us that language code alone is inadequate; regional differences often require different translations for the same language. For example, Chinese (language=zh) requires different translation for China (country=CN) and Taiwan (country=TW). So we need to be able to specify zh_CN and zh_TW as separate locales.

Submissions are reviewed by PhET, and then published to the PhET website. When we “review” a submitted translation we are checking to see that it runs, is complete and the layout is O.K. But, we are not qualified to check the accuracy of translations. We depend on our user population to notify us of problems – similar to Wikipedia’s editing model. So far we have received only a few replacement translations (typically more complete), but no objections to any sim translations.

2.3 Website

The PhET website offers three possible delivery methods for simulations: 1) Run a sim online which involves clicking the “run now” button on the website – requires live Internet connection; 2) Download an individual sim by choosing the “download” option, then the sim is saved to the user’s computer for use at anytime; and 3) Download the PhET offline website. This is an installation program that puts the website and all of its content (except teacher contributed activities) onto the user’s computer. A user then has access to all the sims and supporting material without the need for an Internet connection.

All languages for each sim are available on the website. Access can be found either on each individual sim page or on the “translated sims” page. To deliver translated versions of the sims online, Java sims are run via locale-specific JNLP files, executed by Java Web Start, with translated strings bundled in the JAR file. Flash sims are run via locale-specific HTML files, with

translated strings embedded in the HTML. Flash passes the local to the Flash Player through what is known as FlashVars.

When users download an individual sim, they can choose which language they would like. This is done using local specific JAR files for both Flash and Java sims. The downloadable Flash sims are now wrapped within a JAR file which has the disadvantage of requiring the user's computer to have Java in addition to Flash. We have not received any complaints about this from users so it appears to be an adequate solution. For more detail please see the PhET Translation Technical Concerns document (Malley, 2009). Similarly the PhET offline website includes all the translated versions of the sims which are available as JAR files.

Our current website is written in JavaScript without i18n support. Until recently we decided that the site would remain in English and users would hopefully be satisfied with just translated sims. However, our collaborators at ECSME have need for a fully translated mirror site. In addition we have found several independent groups who have translated the text in our web pages and posted this translated version of the site to their own server. We are very happy to see this level of interest; however, it means that these sites are static and cannot reflect updates to the sims or other materials on the site. So we needed to find a way to make the website also easily translated.

A cost-benefit analysis demonstrated that maintaining and adding necessary upgrades to the current JavaScript site are nearly equal the cost of creating a new Java based using Wicket framework. So we've decided to do our 3rd complete rewrite of the website. This new version will have many new features including a very easy to use translation interface, and it will allow the user to choose their preferred language from the homepage when browsing for sims. One of the challenges is to make sure that when choosing languages, users can view the website with common web browsers in different platforms without any encoding problems. This site should be available to the public in the spring of 2010.

3. Arabic Translation Experiences

Here we discuss the work involved in translation of PhET sims from English into the Arabic language. Our experience again confirms that an expert in scientific materials with bilingual knowledge is needed in order to have accurate and understandable scientific sims. We faced several challenges in translating PhET sims from English into Arabic that are shown to be common to other languages and could apply to any online scientific materials.

First, one of the biggest challenges is to make sure that words used in expressing scientific quantities and expressions are commonly used and understood in scientific materials that are being taught in schools and colleges across the Arabic-speaking countries. However, this has proven to be difficult to do as what is widely used in one country to express scientific quantities might not be used in another country due to different scientific background and educational systems. For example, "acceleration" can be translated in two different ways, which some students are familiar with one, but not the other. To complicate things further, even in the same country, educational systems vary between high school and college level. For example, in Saudi Arabia, math equations and scientific notation are expressed using the Arabic language at the high school level but in the English language at the college level. To overcome this problem, a specific version for each country is created and labeled using a locale that includes the language and country code while a more universal version for use in college is labeled with just the language code.

Moreover, different conventions for expressing units in Arabic exist, and for that reason, abbreviations were mostly avoided in the general versions of the sims, unless it is explained, to make sure variables and units can be understood. One drawback to this solution is that when abbreviations are avoided, strings might be too long to fit in the allowed space in the sims. Again, a possible solution is to create different versions for each locale that uses the specific abbreviations and units used in teaching science in classes.

Since the Arabic language is written from right-to-left, the direction of equations and scientific notation also have to be written right-to-left. Handling this was a challenge when using the Translation Utility as it involves shifting between English and Arabic to embed Arabic text in html codes and between brackets. The mixing of two languages with mixed directions in the same string makes it difficult to follow; however, the ability of testing sims using the Translation Utility saves time and possible layout problems can be avoided. The reverse direction complicates writing

equations from right-to-left in multiple strings as that might reverse order. A single string is preferable for writing equations, however a string that is too long causes other problems.

Chemical symbols (i.e. Na for Sodium) were kept in Roman letters as it is the standard in the Arab world. Greek symbols used in science and math, the most obvious being “ π ”, are used in some cases and other abbreviations for symbols that are commonly used in Arabic textbooks were used in others.

For number formats, Arabic-speaking countries use two different systems: Arabic numerals (0, 1, 2, 3, ...), used in the West, and so-called Hindi numerals or Eastern Arabic numerals (٠, ١, ٢, ٣, ...). Some Arab countries use only Arabic numerals while others typically use the two systems interchangeably. We decided to use Arabic numerals rather than translating to Hindi since nearly all students understand the Arabic numerals and additionally with Hindi it's easy to confuse the zero (٠) and a decimal point (.). While text in Arabic runs from right-to-left, conversely numerals run from left-to-right.

One has to be careful when translating scientific materials to avoid possible misconceptions that might arise from choosing a word that is correct linguistically, but might not reflect appropriate description of the intended scientific meaning. Words like “distance” and “displacement” used in mechanics should be carefully translated to reflect their scalar and vector meaning and to avoid using them interchangeably. This again shows why it is best to have a teacher do the translations.

4. Internationalization of Scientific Material

Many of the challenges that our colleagues from Saudi Arabia face, are common to other languages as well. Languages that are written right-to-left clearly pose complications when translating from a language such as English that is written left-to-right. Different character sets bring with them all sorts of font problems. If a user's computer does not support the specified font, empty rectangles will be displayed for all characters. We incorrectly assumed when we started, that conventions for expressing equations, variables, and mathematical symbols were universal. However, we were pleasantly shocked when we learned that chemical symbols, such as Na for sodium, are universal. So far we have not encountered an educator who does not use the IUPAC (International Union of Pure and Applied Chemistry) system for name and symbol of chemical elements.

5. Conclusion

We have created a Translation Utility for translating PhET Interactive Simulations from English to any other language and a system for delivering these translated sims. We now have a process that works quite well and along the way we discovered many of our initial ideas were flawed. Currently scientists/teachers have contributed a total of 1054 translations in 40 different languages that we host on our website. We learned after several other avenues failed, that translators must be teachers who use the sims in order to achieve precise translation of scientific terminology. Along the way we also discovered a few unexpected things about conventions for expressing chemical symbols, scientific-notation, mathematical symbols, abbreviations and numerals used in scientific materials across the globe.

Acknowledgements

This work would not have been possible without the expert simulations created by The PhET Team, particularly its superb software developers Sam Reid, Chris Malley, John Blanco, Mike Dubson, and Jonathan Olson. PhET is supported by the National Science Foundation, the William and Flora Hewlett Foundation, The Excellence Center of Science and Mathematics Education (ECSME) at King Saud University, JILA and the University of Colorado. ECSME is grateful for the Ministry of Higher Education and King Saud University, Riyadh, Saudi Arabia for their support.

References

- PhET Interactive Simulations (2009). Research page: <http://PhET.colorado.edu/research/index.php>
- PhET Team (2007). The PhET Translation Utility. <http://phet.colorado.edu/contribute/translation-utility.php>.
- Deitsch, A. and Czarnecki, D. (2001). *Java Internationalization*. M. Loukides Ed. Sebastopol, CA: O'Reilly & Associates, Inc.
- Malley, C. M. and Olson, J. (2009). PhET Translation Technical Concerns. http://phet.colorado.edu/phet-dist/publications/Technical_concerns.pdf