International Journal of Advanced Computer Research
ISSN (Print): 2249-7277 ISSN (Online): 2277-7970
Volume-5 Issue-20 September-2015

Using Science Museum Curator's Knowledge to Create Astronomy Educational Content

George Moroni Teixeira Batista[1*], Mamoru Endo[1], Takami Yasuda[1], Mayu Urata[2] and Katsuhiro Mouri[3]
Graduate School of Information Science, Nagoya University, Nagoya, Japan[1]
Graduate School of International Development, Nagoya University, Nagoya, Japan[2]
Astronomy Section, Nagoya City Science Museum, Nagoya, Japan[3]

Received: 27 July 2015; Revised: 23 August 2015; Accepted: 28 August 2015

Abstract

The creation of astronomy educational content is sometimes a difficult task, especially in alternative learning environments such as a science museum. Many people visiting science museums have an interest in astronomy but are not specialists in this area. It should be interesting to find out if museum curators could make the knowledge that is present in museums more relevant to people's daily lives, thus making it easier to understand. General astronomy applications can yield really beautiful and precise sky images; however, the images are usually too complex to non-astronomy specialists or beginners. Therefore, it is difficult to use the applications' output images for educational purposes without spending a lot of time editing them. To solve this problem, the idea in this research study was to create an application that allows museum curators to apply the knowledge they already have through, and more quickly edit the images in an effort to make them more intelligible. After application developers worked with museum curators to develop and test such a system, the curators could create the educational content at least two times faster than they could using regular astronomy applications and image editors, and they were also able to easily create a content database and update the content.

Keywords

Astronomy, e-Learning, content, editing, management, stars simulation, stars observation, International Space Station.

*Author for correspondence
This work was supported by JSPS Kakenhi grant numbers 15K00448 and 15K16097.

1. Introduction

This research study started with the Escola de Linguas Online (ELO) project in August 2007. The ELO project had the objective of creating the electronic learning (e-learning) online platform of the Department of Foreign Languages and Translation of Brasilia University in Brazil. Following the same idea stated by Boggs and Jones [1], "the Internet is a professional development tool for teachers," the project tried to implement information technologies in a way that enabled teachers to use them to develop better multimedia teaching materials and to make the materials more accessible to students. As stated by Batista, Urata and Takami [2] and Batista et al [3], "The system has shown good results, allowing teachers to create interactive multimedia teaching materials, even without knowledge of any programming language." The e-learning platform system allowed the teacher to create dynamic teaching materials—teaching materials that the teachers could update themselves and that could evolve as the teaching environment evolved. The evolving feature is really important because as stated by Jones and Lynch [4] and Koehler and Mishra [5], the educational environment is continually changing. The study results also showed the need for a more improved database, content search engine, and content version management features.

The current research's objective was to implement the dynamic teaching materials concept in an e-learning context other than language teaching, with the goal being to improve the database and search engine to create an application that facilitates the creation of astronomy educational content that is more intelligible for non-specialists. To achieve that, a new project with the Nagoya City Science Museum in Japan was started. As stated by Iwazaki et al [6], the Nagoya City Science Museum has a different theme every month and also a large variety of visitors. As a result, it is necessary to create different versions of content to match the current theme or the correct public profile. The Nagoya City Science Museum was a good choice of museum to use to achieve the research objectives.

A wide range of specialized applications for astronomy can be used to create really precise simulations for star observations. Google Sky, Sky Guide, and Solar Walk are examples of applications that can generate two-dimensional (2D) images or three-dimensional (3D) scenes to observe the stars in 3D space. These kinds of applications are very useful for specialists; however, the images they create can be confusing for non-specialists, especially because many stars that cannot be seen by the naked eye have been displayed. To solve these problems, the idea was to create an application that could read the output data from a simulation and allow the curators to configure the rendering of the output image to create an image that the museum public could understand. Also, the application needed to store the rendering settings so that they could be applied to future images that use the same settings. This paper discusses how the development team applied the Nagoya City Science Museum curator's knowledge to create a Web application that we call SkyNavi and how this application was used to achieve the research objectives.

The rest of this paper is organized as follows. Section 2 describes the system concept. Section 3 explains the system features. Section 4 is about the evaluation tests and meetings with the museum curator. Section 5 discusses the conclusions and future work.

2. System concept

This research study focused on image content that the museum creates to show where and when the International Space Station (ISS) can be seen in the sky like a star. To create this kind of content, the museum curator needs to run a simulation that calculates the position of the stars in the sky on a specific date and at a specific time and location on the earth. Next, based on the results of the simulation, the curator must generate an image that represents the vision of the starry sky on the specified date and at the specified time and location. The problem is that the simulation generates an image that has many more stars than are visible to the naked human eyes, and the curator also needs to add symbols such as arrows, add labels to provide explanations, and highlight certain stars and constellations to make it easier to locate where the ISS will be passing in the sky. Figure 1 shows the raw result of the simulation on the left, and the final output image after the application of the filters on the right.

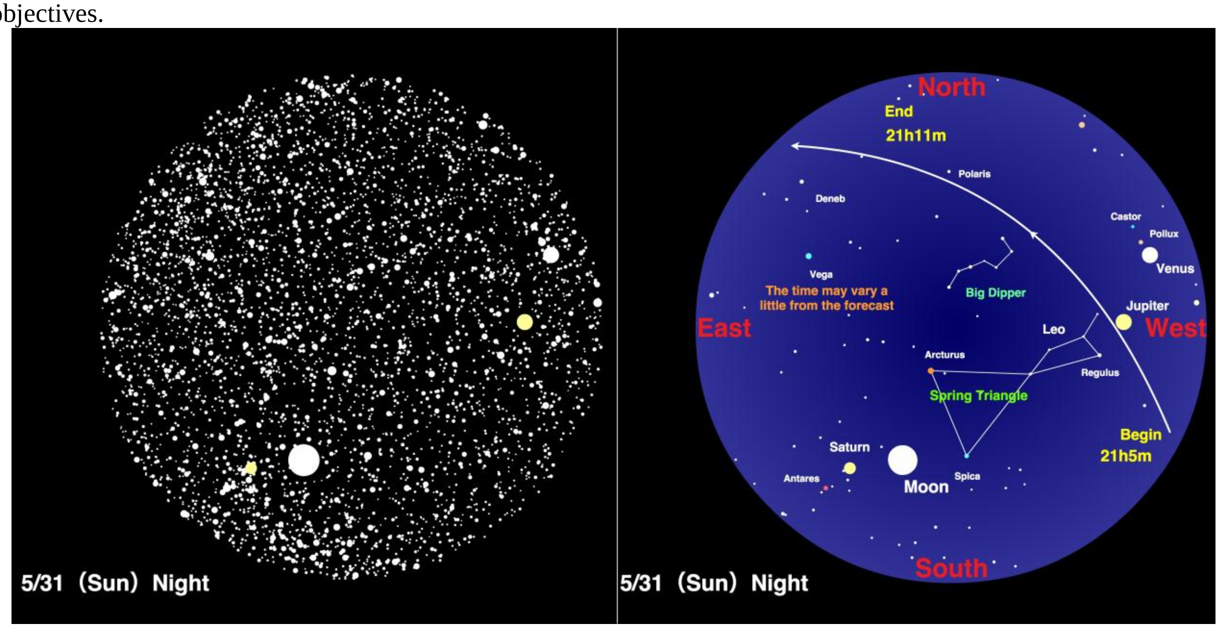


Figure 1: Image showing simulation result on the left, and final content on the right

Prior to this, to create the final version of the content, the museum curator used to edit the output image of the simulation applications with an image editor, usually Adobe Photoshop, to change the colors and sizes of the stars as well as to insert the lines, arrows, and explanation labels. Everything was done by hand, including erasing all of the unnecessary stars and redrawing the stars in their correct sizes. Editing the image was a difficult task and took a lot of time to complete. However, the museum curator knows which stars are visible, which stars are not visible, which stars need to be highlighted, what makes some stars look bigger than others do, and other details necessary to create the final image. Therefore, to solve these problems, it was decided to create an application with an interface to help the curator to apply his own knowledge to the simulation results and generate the final content image.

In the beginning, the idea was to make only one application that could perform a simulation to get the position of the stars and apply a series of filters to adjust the stars' visibility, size, and other necessary settings. However, this kind of simulation is really complex, and the museum curator did not want the algorithm and source code of the simulation to become open-source in a Web application, but the results of the simulation could be shared without a problem. Therefore, it was decided to divide the process using one application for the simulation and another to apply the image filters and then to insert explanation labels and symbols.

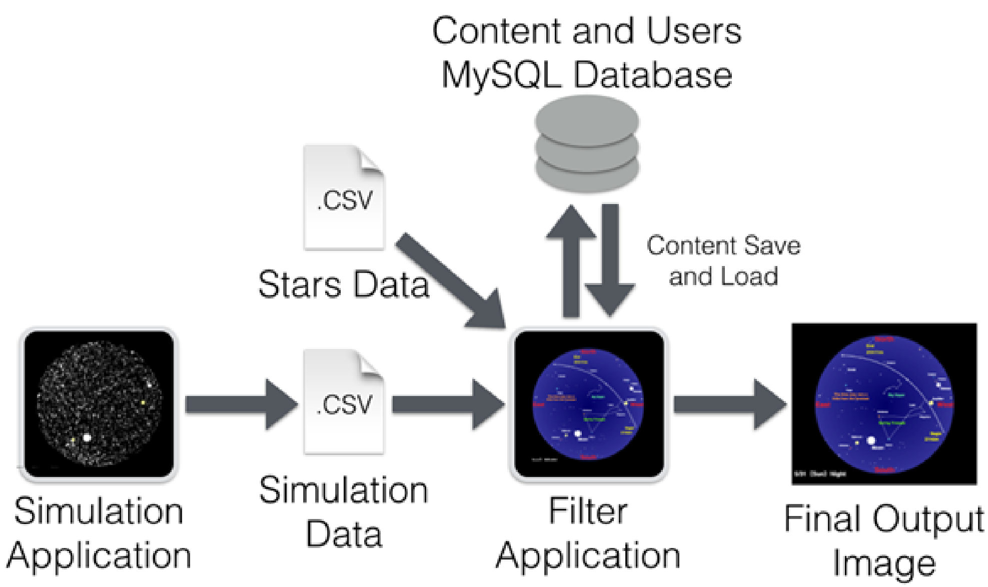


Figure 2: SkyNavi system diagram

Figure 2 shows a complete view of the SkyNavi application system. The system has two main parts: The first one is the source of the simulation data, and the second one is the application that loads the simulation output data and applies the necessary filters to generate the final image. In the current situation, the simulation data were part of an application also created by the research development team.

The simulation application simulates the starry sky, and the result of the simulation is exported as a comma-separated values (CSV) file. The CSV file is uploaded to the Web server for the filter application to access later. The filter application is a Web application created using Adobe Flex framework. It is connected to two databases: One is the CSV file database, and the other database is a MySQL (structured query language) database.

The CSV files have result data from the simulations, and there are also CSV files for stars' information, such as stars' names, constellations, and magnitudes, from the Yale Bright Star Catalog. The MySQL database stores user data and e-learning content settings, content updates, and content versions. After finishing the filter configuration, the user can save the content in the database and also export it as a Joint Photographic Experts Group (JPEG) image; the image format and size depends on where the content will be used. The content stored in the database can be accessed for updates at a later time or be used as a template to create a new one.

When using the system, the user first uses the simulation application. It is an application created using the processing development environment. The application runs in the user's computer like a native application. After one sets the desired date and time, the application outputs a CSV file with the stars' database identification (ID) number and the stars' X and Y positions on the screen. The CSV file is saved with a name that represents the date and time selected in the simulation. Next, the file is uploaded to the server using an FTP client. After the necessary data is uploaded, the user can access the filter Web application and choose the desired simulation data through a list of dates and times based on the available simulation result files in the database.

To use the filter application, first, the user needs to create an account. The account can be created in the application's initial screen. After one creates the account and login, the application loads the stars' information from the CSV database. This information is used to create star objects inside the application; these objects hold the stars' information, such as the stars' database index, X and Y positions, magnitudes, current sizes on the screen, and constellation. When the user selects simulation data, the application matches the ID number in the CSV file list with the star objects to check their positions, and it displays the stars that are in the view range. After the application positions and displays all of the necessary stars, the user can begin to set the filters.

The filters are basically the application of the curators' knowledge of which stars are visible, how they are visible, and if there is some kind of influence among them—for example, one star is brighter; therefore, it is more visible and is represented by a bigger point on the screen, thus showing the contrast among the stars.

However, instead of editing each star by hand, for example, the curator knows that the visibility of the stars is related to the stars' magnitudes and that the stars with magnitudes higher than 3 are not normally visible; therefore, the curator can set the magnitude filter to show only stars with magnitudes equal to or less than 3. Also, the curator can set the contrast filter to decrease the size of a star in relation to the brightest star being displayed based on the difference between their magnitudes. Filters are also available to adjust the overall star size scale and to show or hide star names. Section 3 will provide a more detailed description of the system features.

After setting the filters and inserting labels and symbols to provide explanations, the user can save the content in the MySQL database; basically, the application stores the filter settings and the inserted labels and symbols in the database. The creation of this database is important because, as stated by Zhang et al [7], the reutilization of existing contents is really important for the creation of new content in any area. This feature can speed up the creation of content because the user will not need to create every piece of content from sketch. It also allows for the creation of different versions of the content oriented to different members of the public, enables content updates, and allows one piece of content to be used as the template for a new piece content using the same filters, or it allows for the application of the same filters to a different simulation result. When saving the content, the user can insert tags that can be used to describe the content and also be used as search keywords.

The output images from a specialized astronomy application are really good, precise, and very helpful for advanced astronomy study or research. However, these images are too complicated to use for educational purposes for beginners or for museum visitors who are not specialists in astronomy. The application of the filters to the simulation data is intended to create final images that are most like what people really see when they look at the sky with their naked eyes; make it easy to understand; and show how people can find the right position and direction for spotting the ISS at a forecasted time.

3. System main features

This section will provide a detailed explanation of the system main features, why the features were created, how the features are intended to represent the Nagoya City Science Museum curators' knowledge, and how the users are supposed to use them and the interface created for the features.

Having the development team enter the user's context helps the team members to work together with the user to create something new and original—something that can really fulfils the user's necessities, as stated by Vianna et al [8]. This collaborative work helps the development team to incorporate the museum's curators' knowledge into the system functions and to make sure that the system does what it is meant to do.

3.1. Star position simulation

To start creating the e-learning content, the user first needs to get the stars' simulation data related to the desired date and time. In the current situation, the part of the system that made the simulations was a different application because the museum curators did not want the simulation algorithm and source code to become open-source in a Web application; therefore, the simulation application ran offline as a native personal computer (PC) application in the user's PC. When using this application, the user only needs to set the date and time, making the clock go forward or backward using the keyboard keys ">" and "<," respectively. When the desired time is set, the user can push the "0" key to output the data in a CSV file.

The CSV file is named with the date and time set in the simulation in the format YYYYMMDD_HHMMSS.csv; for example, if the simulation date is set to May 31, 2015, at 21 hours, 8 minutes, and 10 seconds, the name of the file will be 20150531_210810.csv.

The content of the file is a simple list of the stars that were inside the simulation screen with their database index, X and Y positions, and magnitudes for normal stars. In the case of the Sun, Moon and other planets in the solar system, because they have neither a database index nor magnitudes, the index is replaced by the "#" character, followed by a solar system index number: 0 for Sun, 1 for Moon, 2 for Mercury, 3 for Venus, 4 for Mars, 5 for Jupiter, 6 for Saturn, 7 for Uranus, 8 for Neptune, and 9 for Pluto. The last two numbers are the X and Y positions.

After the getting the output file, the user needs to upload it to the server by using, for example, a File Transfer Protocol (FTP) client. Unfortunately, in the current version of the system, neither the simulation application nor the filter application had an upload function for the simulation files, but it did not seem necessary during the tests. After the upload, the simulation data are ready for the filter application to use.

3.2. MySQL database

The system has a MySQL database for user and content data. The database also stores tags used to identify and describe content as well as the relationship between pieces of content. The content relationship can show when a piece of content was created based on other content, using a piece of content as a template or by combining two pieces of content. The database also has tables for storing the explanation labels on the content.

Therefore, the user can reuse labels if necessary, and there are also tables for the constellation data, which are needed to draw the lines between stars in a constellation in the right order. Before creating content with the filter application, the user needs to create a user account. The account can be created on the first screen of the filter application in the login panel.

The user only needs to input his/her email address, password, first name, and last name as well as choose a language. After creating the account, the user can log into the account using his/her email address and password; the interface language can be changed after login, too.

When saving a piece of content, an explanation label, or a constellation, the user can tag it with keywords to describe it and also help to find it afterward using a search engine.

For the tag system, the system has a table for storing the tags, and three other tables to list with the tags are related to the content, label, and constellation. The tables simply have entries with the tagged content database ID, the tag ID, the ID of the user who tagged the content, and the date of the entry.

The table for the content stores data such as when and who created it, the name of the piece of content, the filter settings, inserted symbols and labels, highlighted constellations, and the last time the content was viewed.

The table also has three columns for storing information about the relationship between the pieces of content when necessary. When the user creates a piece of content using another content piece as a template, the first relationship column stores the ID number of the content piece used as a template. Every time a piece of content is updated, there is a new entry in the database for the update. The second relationship column stores the ID number of the first version of the content. When a piece of content is created based on another content piece, the database also stores the ID number of the original version of the content in the third relationship column.

These three relationship columns allow the system to track how each piece of content changed over the time, all of the updates, all of the different versions of the content, and the updates of the versions.

3.3. Loading simulation data

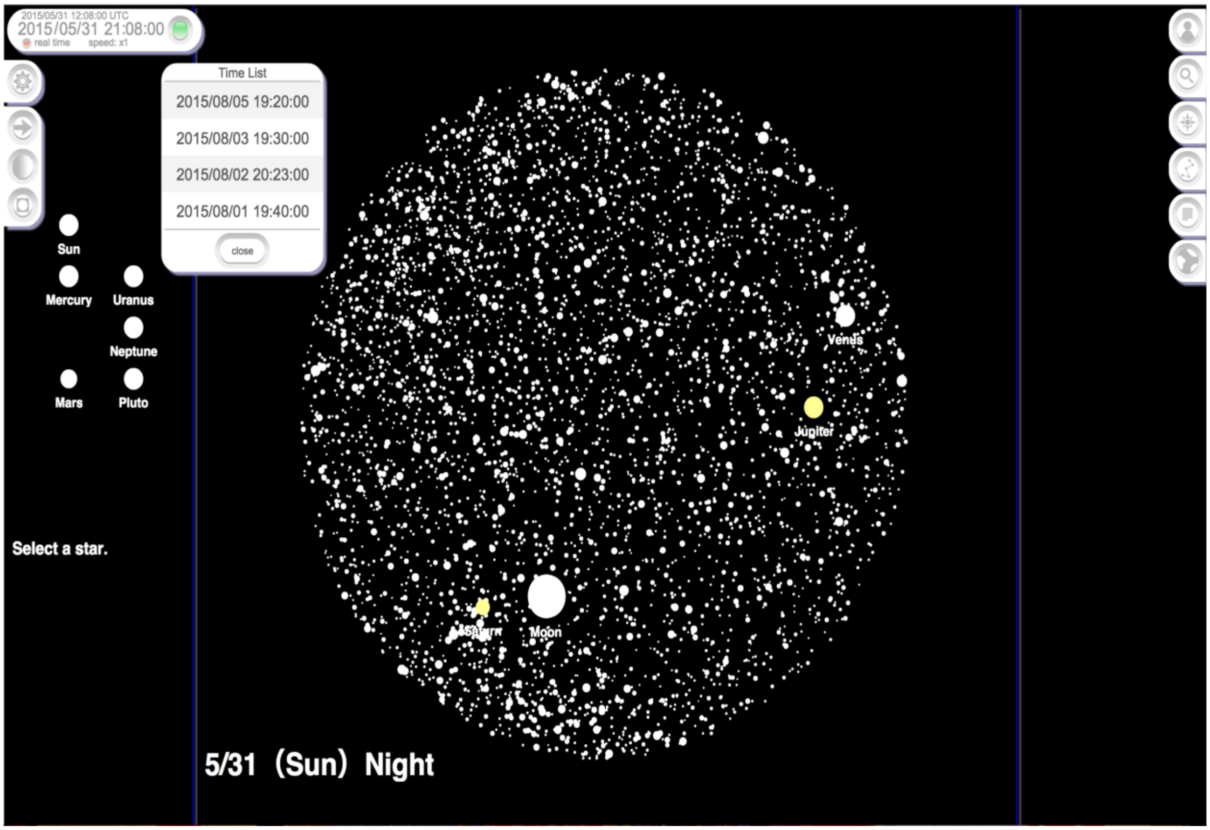


Figure 3: Clock and time list used for loading simulation data

After login, the user interface of the filter application appears. The first thing the user needs to do is set the date and time using the clock on the upper left side of the screen. Originally, the idea was to allow the user to set any date and time; therefore, the application could do the simulation and show the stars on the screen. However, because the simulation function has been moved to the other application, the user only needs to click the green button that appears when the mouse cursor is near the clock and choose the desired date and time from the time list that appears, as shown in Figure 3.

After choosing the date and time, the clock is automatically set, and the simulation data related to the selected date and time are loaded. The simulation data have a list of the stars that are in the view range and their positions. After loading these data, the application displays all of the stars in basically the same way, but there is a slight difference in the sizes based on the stars' magnitudes. The planets and the moon each have a predefined size, and the moon phase is calculated based on the date and time of the simulation.

3.4. Filters

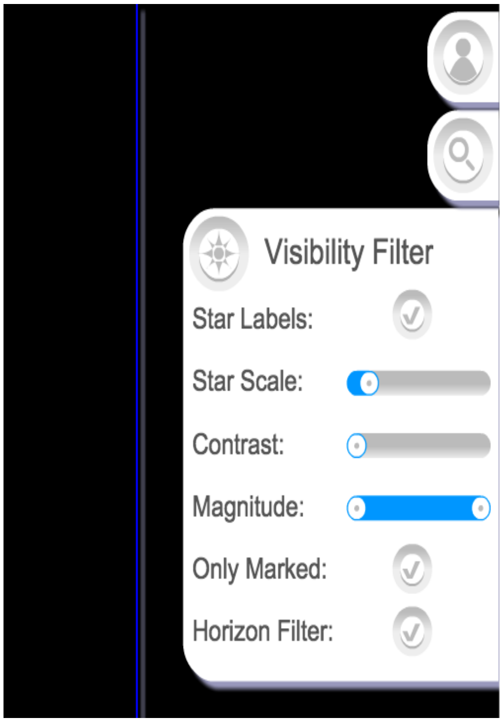


Figure 4: Filters panel

After the simulation data are loaded, the user can begin to apply the filters and insert the explanation labels and symbols to create the e-learning content. As stated by Alexander [9], "the aim of all education initiatives (regardless of the medium used), is to make it possible for students to learn;" therefore, the main idea of the filters is to simplify the image created by the simulation with the purpose of making it intelligible for non-specialists and also to make it closer to what people see when they look at the sky at night, considering external factors such as weather, the current season, and light pollution. Figure 4 shows the filters' panel. The filters in the panel affect all visible stars on the screen, but it is also possible, and sometimes necessary, to make small adjusts to individual stars or to a smaller group of stars.

3.4.1. Star label filter

The star label filter is a simple filter that shows or hides the stars' name labels. This is useful when the number of visible stars is low. The user does not need to turn the star labels on or off one-by-one; he/she can simply select all of the stars first and then change the star filter setting.

3.4.2. Size scale filter

The size scale filter changes the overall scale of the stars' sizes. The filter has a slider control that ranges from 0 to 100; the selected value multiplies the original star size calculated based on the star magnitude. This filter is used to adjust the scale of the stars in the final image if the user needs stars to proportionately look bigger or smaller depending on what he/she is explaining or wants to show.

3.4.3. Magnitude filter

This filter is controlled by a slider control that ranges from -1.46 to 7.96. The slider has two values: a minimum value and a maximum value. Basically, the user sets the values in a range between -1.46 and 7.96 using the minimum and maximum sliders. All of the stars with magnitudes lower than the minimum slider value magnitudes higher than the maximum slider value are not displayed. This was the first filter for which the museum curators asked and is probably the most important because it defines the range of the visible stars.

Before, the curators had to separate the visible stars one-by-one in the image editor, but now, they only need to move the sliders to the desired values, and the system separates the stars automatically. Because the stars’ visibility is directly related to their magnitudes, and because the curators know the magnitude range that can be seen with the naked human eye, it is much easier and faster to simply select the minimum and maximum magnitude values than to have to check the magnitudes of the stars one-by-one to determine if the stars are visible.

3.4.4. Contrast filter

The contrast filter is used to create a contrast effect among the stars. The filter is controlled by a slider control that ranges from 0 to 1; the selected value defines the intensity of the contrast effect. This filter is used to simulate the effect of the brighter stars obfuscating the other stars. Without this filter, all of the stars appear to have almost the same size, even with their magnitude differences. The contrast filter works based on three values: the contrast intensity, the minimum magnitude, and the maximum magnitude set in the magnitude filter. Basically, the system retrieves a normalized value of the star magnitude inside the magnitude minimum and maximum range, finds the contrast level by multiplying it by the contrast intensity, and then multiplies the size of the star by 1 minus the contrast level.

The result is, the stars with magnitudes equal to the maximum magnitude value set in the magnitude filter remain the same, and the stars with magnitudes lower than the maximum become smaller based on the how far they are from the maximum magnitude value.

This filter is also very important to the system. Before, museum curators needed to spend a lot of time making changes to the star sizes one-by-one, by hand, in the image editor because they needed to check every star's data to see how much bigger or how smaller a star should be compared to the other stars. Now, because the system already has these kinds of data, it can automatically calculate the sizes of the stars based on the filter settings.

3.4.5. Horizon filter

The horizon filter is a really simple filter; it has only an on and off button. When turned on, the filter hides the stars that are less than five degrees higher than the horizon line. Those stars are usually so low in the sky that they cannot be seen because of the terrain or the city buildings, for example. In the case of this kind of content, the horizon line means the border of the blue circle.

3.5. Star settings

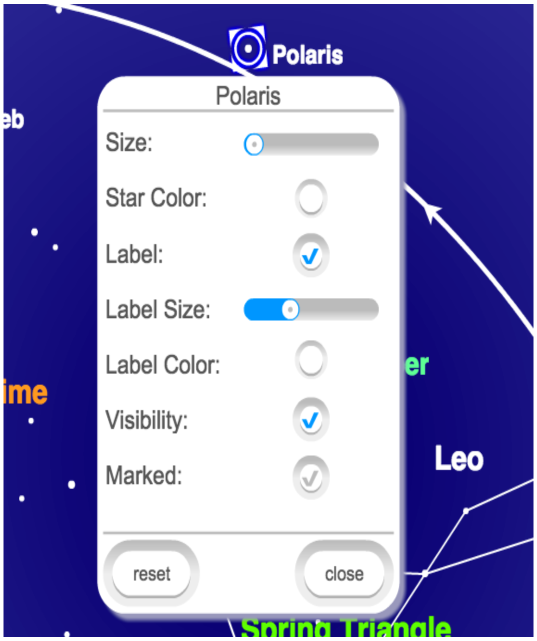


Figure 5: Star settings menu

The filters can help the user by applying changes to all stars at once, but sometimes some specific details need to be applied to specific stars. For that kind of case, the application also has a star menu that can be used to change one specific star or a group of stars (Figure 5). The menu has controls for adjusting the star size and color and for showing or hiding the star name label, the size and color label, and the star itself. The filters are usually adequate for adjusting the star size, but for settings such as the color or to display the star name, it is complicated to have an automated feature because these kinds of settings do not have specific patterns, and they also depend on the kind of content the user wants to create.

3.6. Moon and planet settings

The moon and the other planets in the solar system also have a menu similar to that of the other stars, but in the case of the moon and the planets, they are not affected by the general filter. Therefore, all of the necessary settings need to be adjusted using individual menus.

The simulation application can calculate the positions of the moon and the planets, but defining how they are visible from Earth is a really complicated task because their magnitudes change as they move in relation to Earth. Therefore, the museum curators in the current research study opted to do this manually themselves when necessary.

The moon menu has some additional options for manually setting the moon phase and rotating the moon. In the current situation, the simulation application only gave the position of the Moon, but sometimes it needed to be rotated. The Moon could also be dragged; this feature was useful because sometimes the moon got in front of a star that was an important visual reference in the content.

Therefore, the museum curators asked to make the moon draggable to help to make the image easier to understand, even if that meant the moon position would not exactly match its position in the real sky.

3.7. Constellations

The constellation database is being created as the museum curators use the application.
They are registering the constellations as needed when creating the content. When the user needs to show a constellation that is not yet registered, he/she

can register it by creating a new star path object using the filter application.

The star path object is a simple object with a list of the stars that are used as references for drawing the lines of the constellation. To create a new star path object, the user only needs to choose the reference stars in order. The order is important because the application will follow the list order for drawing lines between the stars. After choosing the stars, the user needs to open the path list side panel on the right side of the screen and click on the "add lines" button or "add polygon" button. The difference between the two is that the "add polygon" button will make an extra line connecting the last star of the list with the first one, thus making a closed polygon.

After adding the star path to the content, the user can make some changes to the path using the path menu (Figure 6). The user can, for example, change the lines' colors and show or hide the constellation name label.

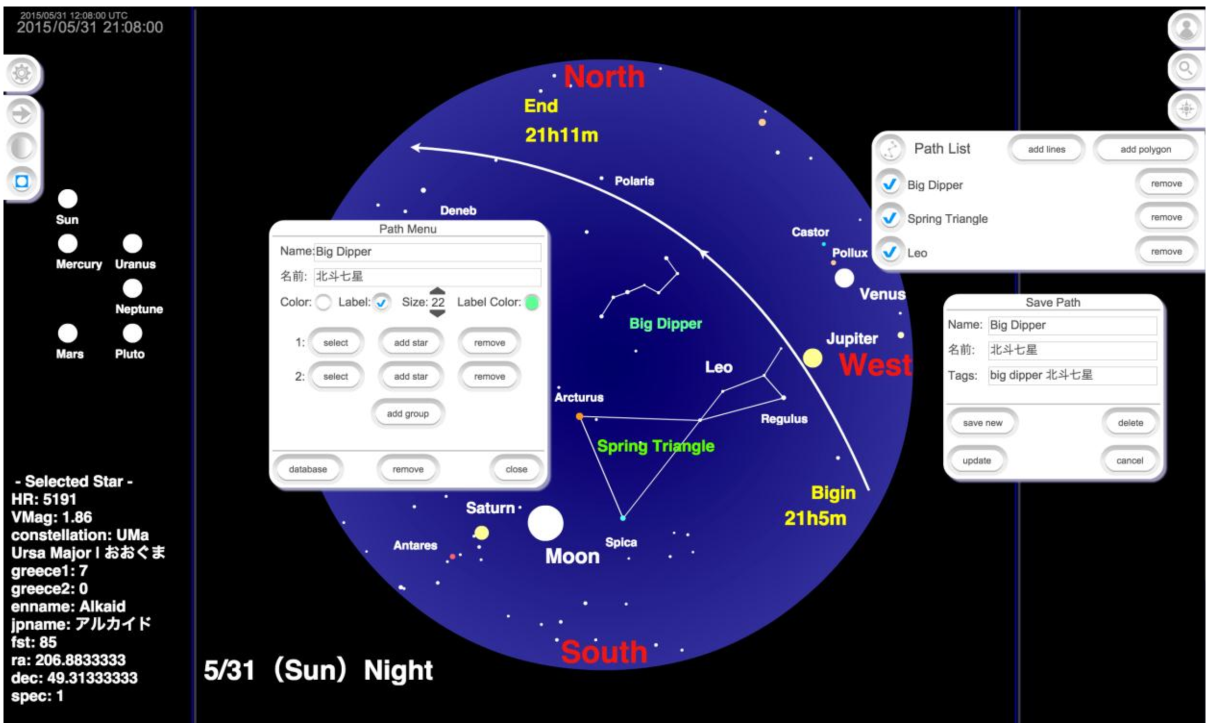


Figure 6: Constellation menus

The user can also register the created path in the constellation database using the menu. Because the constellation lines are drawn using the stars as references, the same database entry can be reused in any situation in which the constellation is visible, regardless of the position of the stars on the screen. In other words, the same star path object can be used in any content that shows the constellation it represents. Before, curators needed to draw the constellation lines by hand in the image editor every time; now, they only need to set the constellation once.

3.8. Labels and arrows

To insert the necessary explanation about when and where the ISS can be seen in the sky, the user can insert text labels and draw lines, curves.
The text labels are usually simple text messages used to explain the time interval when the ISS can be seen, and an arrow is used to show its trajectory in the sky. The application has four default labels that indicate the four cardinal points, and there is also another default label for the date on the bottom left side. These default labels are always present on all content. Other labels appear with a certain level of frequency;

however, because they can change a little depending on the content, these labels are stored in the MySQL database as templates. When needed, the user can load the template label from the database and only change the necessary part.

3.9. Background

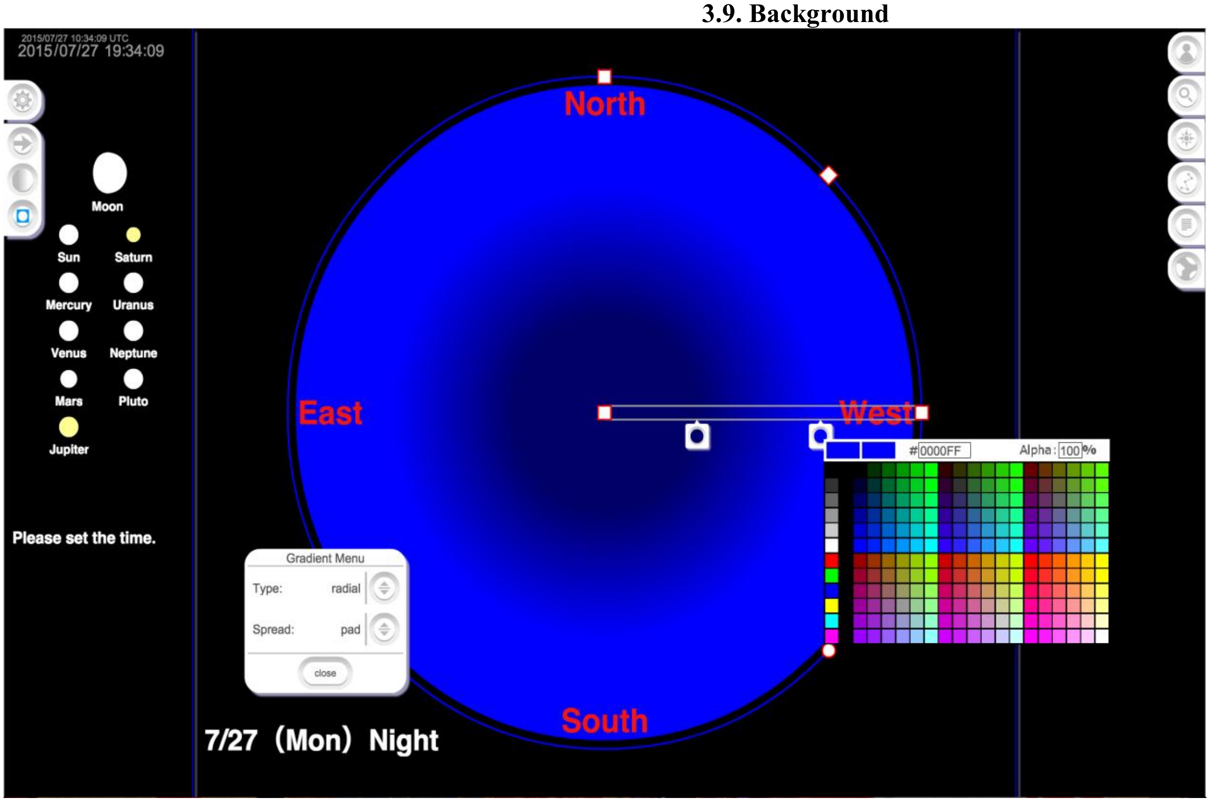


Figure 7: Background gradient controller

The default background of the content is completely black, but it can be changed using the background gradient controls. The controls for the background color can be shown by clicking the gradient button on the left side of the screen (Figure 7). Because the final image is a circle, the default gradient type is radial, but it can be changed to linear. The user can add more colors by clicking inside the gray rectangle and dragging the colors to change the colors' positions and order. The red circle and squares can also be dragged to adjust the rotation angle, width, height, and scale of the gradient. The background color can be changed to mimic the sky's visual condition as expected by weather forecasting and other visual factors.

3.10. Content save and load

After the user has finished configuring the filters, inserting the labels and arrows, setting the background color, and completing other possible necessary tasks with the content, he/she can save the content in the database. In addition to the explanation labels and the constellations, the overall content settings can be stored in the database. The application saves all of the filters' settings, all inserted labels and arrows or lines, and the individual settings of all stars, the moon, and the planets as well as the background gradient configuration. When saving, the user also needs to insert the content name and then tag the content with keywords. The database entry of the content also has columns for the content author, date of creation, and content relationship reference as explained in Section 3.2. To access a piece of content stored in the database, the user needs to use the search panel on the upper right side of the screen. After inserting one or more keywords and pressing the search button or the enter key, the system shows the results of the pieces of content that match the inserted keywords, if any. The search results also include explanation labels and constellations;

therefore, the user can use the same search box to search for any kind of content in the database.

The search filter organizes the results, showing the newer content pieces first. The user can also search for newer or older updates of a given piece of content by clicking the updates button under the content name in the results list. A new panel will open showing all of the updates of the selected content. The versions button will also open a new panel showing all of the content pieces created using the selected piece of content as a template. That allows the users to have access to all updates and to all versions of all of the content pieces and also to understand the relationship between them and examine how the content pieces have changed over time.

4. Tests and evaluation

To evaluate the system, a series of tests and meetings were conducted with the Nagoya City Science Museum curator beginning on November 7, 2014. During the meetings, the development team discussed with the museum curator problems concerning the creation of the content and possible ways of solving them. The main problem that the curator faced was the time spent on creating the content. To solve this problem, the development team created features in the system that automated some of the content-creating processes that the curator had to do by hand using an image editor—for example, drawing or erasing stars. In the first two months, the tests were focused on the application's interface. After the museum curator explained the desired system features, the development team needed time to adjust the interface and to fix problems because the curator usually had time to meet the development team in person only once per month, and most of the communication was done by email. During the tests, the curator who was the main tester already had some knowledge of programming and image editing. This helped to foster communication and a mutual understanding during the development process because both sides knew their necessities and limitations. However more tests still need to be done for the interface, as stated by Krug [10] and Millani [11]. Those curators who do not have this kind of specialized knowledge about image editing will probably try to use the application in different ways, and because even the order of some processes is important in the application, they may have some problems with the interface.

After the main interface was finished in the current research study, the curator stated that he needed 30 minutes to create the first image using the system. For the second one, he needed only five to ten minutes to create a new image from scratch. When creating images using Adobe Photoshop, the curator spent about 20 minutes per image, as the images could not be combined or used as templates for new ones. Every image needed 20 minutes; now, only five to ten minutes are needed to create completely new images, and the process of combining or using other content pieces as templates to create images is even faster. That makes the process using SkyNavi at least 50% faster per image. Table 1 shows a time comparison between the content creation process done using Photoshop and SkyNavi in a situation where multiple similar images needed to be created to show the sky of four consecutive days, all the images have size of 2400x2100 pixels.

Table 1: Time comparison between processes involving multiple images

Image	Time with Photoshop	Time with SkyNavi
first	20 minutes	10 minutes
second	20 minutes	5 minutes
third	20 minutes	5 minutes
fourth	20 minutes	5 minutes
Total time	80 minutes	25 minutes

The implementation of the content database and search engine in the current study was also an important factor in improving the speed of the content-creation process. The database allowed the user to store the content configurations in the system database and to reuse them as templates to create new content, combine two or more content pieces, and reload and update content. This was not possible before because the curator had only the finished images, and in order to create new content for a different date and time, even if the visible star and explanation labels were the same, a new simulation output image was necessary to get the correct positions of the stars. Therefore, the curator needed to edit the new image from the beginning again. Now, the curator can load the new simulation data and just load and apply the desired filter configuration and explanation labels from the database, a process that should take seconds instead of minutes.

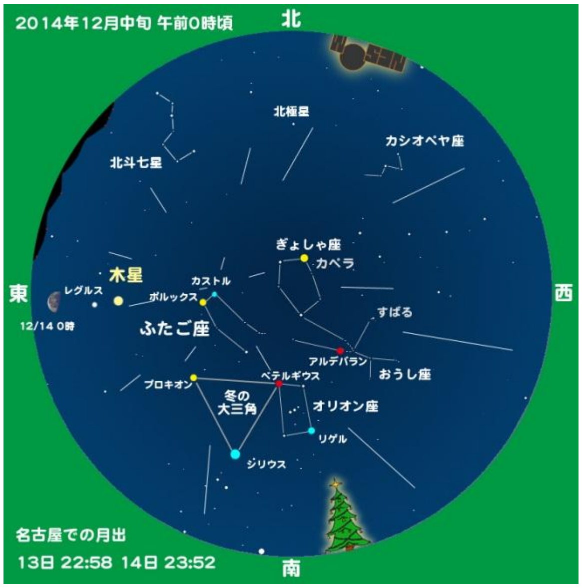


Figure 8: Christmas-theme content

During the tests, the only content that the curator could not create using SkyNavi only was content for an observation near Christmas Eve (Figure 8). It was not possible because in the specific content piece, the curator wanted to use city Christmas decorations as references to help to orient the sky observation. For this, it was necessary to insert some extra images on the horizon line to show the direction of the city's Christmas tree and the Nagoya City Science Museum. In this case, the curator created the base observation content using SkyNavi and edited the final image using Photoshop afterward to insert the images of the Christmas tree, the museum and the mountains.

The system can also be used for other research purposes; for example, most of the study of astronomy is done by observing images of the stars in the sky. This kind of observation can be done using normal images; however, the implementation of filters can facilitate the observation of specific stars or a group of stars over time. Unfortunately, in the current research study, the simulation function needed to be removed from the filter application. That made the overall process of creating content a little slower because the user needed to create the simulation data and upload them to the server before actually beginning to create the content. However, it also opened the door to using simulation data from different sources, applications, or databases. Especially for the developer of the simulations, it should be interesting to have an application that allows them to quickly compare different simulation results and different data sources. This could also allow for the creation of different versions of the content based on different simulation data sources.

The content created using SkyNavi are usually posted on the Nagoya City Science Museum website in the area featuring study and astronomy news on the ISS. As mentioned earlier, the content is used to explain when and where the ISS can be visible from Earth. The final content image is very similar to the images previously created using Photoshop; however, the creation process became much faster and also allowed the curator to create a content and template database and to easily update the content when necessary.

5. Conclusions and future work

This paper discussed how users' knowledge can be applied in the development of an application. The development team working side-by-side with the users can allow the team members to better understand the users' situation and the context in which the application will be used. Testing the application with the museum's curator during the application development process in the current research study allowed the development team to make changes to the application in order to better fulfil the museum's needs and also the research objectives. Changes such as dividing the system into two applications had both bad and good consequences; for example, the overall process became slower than expected. However, it also opened the system up to new possibilities that had not been enabled before.

It is accurate to say that the system had a good evaluation, as the amount of time necessary to create the content was the main problem for the user, and with SkyNavi, this time was cut in half at least. Some content still cannot be created using just SkyNavi. However, the part of the content that cannot be achieved using SkyNavi has no relationship with the curators' astronomy knowledge. It is related to very specific cultural aspects of the region and local environment that were beyond the focus of this research study. The system also allowed the team to create and test an improved version of the database and search engine used in the dynamic teaching materials system. The improvement allowed the user to track the evolution of a piece of content and see how it changed over time; it was also used to help to create other content.

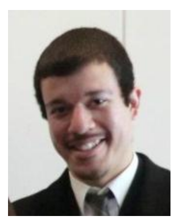
The improved database and search engine are an important step in the next part of the research study. The possibility of tracking the evolution of the content and seeing how it changes over time can also help the user to understand why the content changes over time. For example, if a user uses tags or comments to describe what is new in the updated version of the content and why the updates were made, maybe other users can learn through his/her experience and get some advice about editing content. As stated by Jones, Sharonn and Power [12], the creation of this kind of community learning among system users can allow them to evolve and to become better content creators as they use the system and learn from other users' experiences.

Acknowledgement

This work was supported by JSPS Kakenhi grant numbers 15K00448 and 15K16097.

References

[1] E. Boggs and D. Jones, "Lessons learnt in connecting schools to the internet", Australian Educational Computing, Vol. 9, No. 2, 1994, pp.29–32.
[2] G. M. T. Batista, M. Urata, T. Yasuda, "The dynamic teaching materials system: a way to make teaching materials evolve", International Journal of Knowledge and Web Intelligence, Vol. 3, No. 4, 2012, pp.343-360.
[3] G. M. T Batista, et al. "Using Patterns to Guide Teachers and Teaching Materials Evolution", The Seventh International Conferences on Pervasive Patterns and Applications, France, March 2015, IARIA, 2015, pp.8-14.
[4] Jones, D. and Lynch, T. "A model for the design of web-based systems that supports adoption, appropriation, and evolution", Murugesan, S. and Deshpande, Y. (Eds.): Proceedings of the 1st ICSE Workshop on Web Engineering, Los Angeles, 1999, pp.47–56.
[5] Koehler, M. J., and Mishra, P. "What is technological pedagogical content knowledge?", Contemporary Issues in Technology and Teacher Education, Vol.9, No. 1, 2009, 60-70.
[6] K. Iwazaki, et al. "Possibility and the Trial based on "connection" between the Science Museum and Universities or Visitors: Development of Exhibitions for the 50th Anniversary Event of the Nagoya City Science Museum", Journal of the Japan Information-culture Society, Vol.20, No. 1, pp.10-17, May 2013.
[7] J. Zhang, et al. "Managing Presentation Slides with Reused Elements ", International Journal of Information and Education Technology, Vol. 6, No. 3, March 2016.
[8] M. Vianna, et al. "Design thinking Business innovation", 1st ed., MJV Press, Rio de Janeiro Brazil, 2012.
[9] S. Alexander, "E-Learning Developments and Experiences", Technological Demands on Women in Higher Education: Bridging the Digital Divide, Cape Town, February 2001.
[10] S. Krug, "Don't Make Me Think! A Common Sense Approach to Web Usability", 2nd ed., New Riders, Berkeley, California USA, 2006.
[11] B. Gillani, "Learning Theories and the Design of E-Learning Environments", United States of America, University Press of America, 2003.
[12] D. Jones, S. Sharonn, and L. Power, "Patterns: using proven experience to develop online learning", Proceedings of ASCILITE '99, Responding to Diversity, Brisbane: QUT, 1999, pp. 155–162.



George Moroni Teixeira Batista, born in Brazil in 1985. In 2012, he got his master degree in the Graduate School of Information Science of Nagoya University in Japan, and now he is a doctoral student in the same institution. His main research interests are e-Learning and interface design.
Email: tenno.kun@gmail.com

Mamoru Endo, born in Japan, 1974. Received his Ph.D. from Nagoya University in 2003, and in the same year became a lecturer in Chukyo University. Now is an associate professor in Graduate School of Information Science, Nagoya University.

Takami Yasuda, born in Japan, 1959. Became doctor of Information Engineering, Nagoya University in 1987, in 1993 became associate professor of School of Informatics and Sciences, Nagoya University, and since 2003 is professor of Graduate School of Information Science at the same university.

Mayu Urata, born in Japan. Received her master and doctor degrees from Graduated School of Information Science, Nagoya University in 2008 and 2011 and respectively. In 2011 also became assistant professor in Graduate School of International Development, Nagoya University.

Katsuhiro Mouri, born in Japan, 1964. Received his master and doctor degrees from Nagoya University in 1988 and 2011 respectively, and currently is curator and chief of the astronomy section of the Nagoya City Science Museum.