Chapter 4

The WeCurate Application

Matthew Yee-King, Dave de Jonge, Roberto Confalonieri, Nardine Osman, Mark d’Inverno, Carles Sierra, Leila Amgoud, and Katina Hazelden

Multiuser museum interactives are computer systems installed in museums or galleries that allow several visitors to interact together with digital representations of artefacts and information from the museum’s collection. WeCurate is such a system, that allows users to collaboratively create a virtual exhibition from a cultural image archive. It provides a synchronised image browser across multiple devices to enable a group of users to work together to curate a collection of images. WeCurate uses electronic institutions to coordinate and synchronise the interactions between individuals, and it relies on agreement technologies (such as argumentation and computational social choice) for collective decision making. This paper provides an overview of the WeCurate application, describes its underlying electronic institution, and presents a brief introduction to its collective decision making mechanism.

Matthew Yee-King
Goldsmiths College, London, e-mail: m.yee-king@gold.ac.uk

Dave de Jonge
Western Sydney University, Sydney, e-mail: d.dejonge@westernsydney.edu.au

Roberto Confalonieri
IIIA-CSIC, Barcelona, e-mail: confalonieri@iiia.csic.es

Nardine Osman
IIIA-CSIC, Barcelona, e-mail: nardine@iiia.csic.es

Mark d’Inverno
Goldsmiths College, London, e-mail: dinverno@gold.ac.uk

Carles Sierra
IIIA-CSIC, Barcelona, e-mail: sierra@iiia.csic.es

Leila Amgoud
Toulouse Institute of Computer Science Research (IRIT-CNRS), Toulouse, e-mail: Leila.Amgoud@irit.fr

Katina Hazelden
Freelance graphic designer, UK, e-mail: kat9@mac.com
4.1 Objectives

In recent times, high tech museum interactives have become ubiquitous in major institutions. Typical examples include augmented reality systems, multitouch tabletops and virtual reality tours [11, 14, 20]. Whilst multiuser systems have begun to appear, e.g. a 10 user quiz game in the Tate Modern, the majority of these museum interactives do not perhaps facilitate the sociocultural experience of visiting a museum with friends, often being designed for a single user. At this point, we should note that mediating and reporting the actions of several ‘agents’ to provide a meaningful and satisfying sociocultural experience for all is challenging, requiring multiple criteria decision making. Another trend in museum curation is the idea of community curation, where a community discourse is built up around the artefacts, to provide different perspectives and insights [19]. This trend is not typically represented in the design of museum interactives, where information-browsing, not information-generation is the focus. However, museums are engaging with the idea of crowdsourcing with projects such as ‘Your Paintings Tagger’ and ‘The Art Of Video Games’ [12, 6]. Again, controlling the workflow within a group to engender discussion and engagement with the artefacts is challenging, especially when the users are casual ones as in a museum context.

In this chapter, we describe WeCurate [22, 2], an image browser for collaboratively curating a virtual exhibition from a cultural image archive – group curation. The WeCurate application allows users to synchronously view media and enables negotiation about which images should be added to the group’s image collection. Further, it accelerates the navigation of extensive museum databases and provides a platform for sociocultural experiences, combining the actions of autonomic agents and users to facilitate decision-making. WeCurate is tasked with establishing the users’ presence in the shared experience by enabling communication around the deconstruction and appropriateness of the media, representing social proxies, and making agent and group members’ actions manifest to everyone.

WeCurate uses a multiagent system to support community interactions and decision making and an electronic institution [9] to model the workflow. The aim of this work is not only to make use of agent technology and electronic institutions as a means to implement a multiuser museum interactive, but also to relate agent theory to practice in order to create a usable system that will help us to answer the following research questions. Do agent technology and electronic institutions enable users to share online experiences in a way that was not possible before? Are decision making capabilities and aggregation operators defined in the literature suitable for enhancing the experience of users? Can we create a user interface which represents the state of the underlying system to the user in a meaningful way? Can we create a sense of social presence between the users of the system?

To this end, we specify a community curation session in terms of the scenes of an electronic institution for controlling community interactions. We make use of a multimodal user interface which directly represents users as agents in the scenes of an underlying electronic institution and which is designed to engage casual users in a social discourse around museum artefacts. One key underlying technology is then
used to support system’s and users’ decisions by means of personal assistant agents equipped with different decision making capabilities.

In WeCurate, the automated decision making of the agents is undertaken by different models [8]: preference aggregation [23], multiple-criteria decision making [1, 17] and voting. Preference aggregation allows having a quick understanding whether the users consider an image as interesting or not. To this end, each user expresses an image preference and a collective decision is achieved by preference aggregation. Multiple-criteria decision making is a collective decision taken as a result of a negotiation protocol. Users exchange image arguments according to an argument-based multiple criteria decision making protocol. Voting takes the majority of the agents’ decision into account. For further information on the automated decision making technology, we refer the interested reader to [8].

4.2 Illustration

In this section we present the user interface of the WeCurate system [13]. The interface is an animated user interface which has a separate screen for each scene in the electronic institution. The three main scenes are presented by the three main views in Figures 4.1, 4.2, and 4.3.

1. **Selection view.** The purpose of this view is to allow a quick decision as to whether an image is interesting enough for a full discussion. Users can zoom into the image and see the zooming actions of other users. They can also set their overall preference for the image using a simple like/dislike slider, which also shows the preferences of other users in the group. The view is shown in Figure 4.1.

2. **Forum view.** If an image is deemed interesting enough by the agents, the users are taken to the forum view, where they can engage in a full discussion of the image. Users can add, delete and weight tags, and can see the actions of the other users so they have a sense of what others in the group are interested in. They can also view images that were previously added to the collection. The view is shown in figure 4.2.

3. **Voting view.** In this view, the users can vote on whether they wish to store the image to their collection. Here, the decision is made to add an image to the group collection, or not. The view is shown in Figure 4.3.

4.3 The EI Specification

In this section we show the underlying electronic institution (EI) specification [9] of the WeCurate application.
Fig. 4.1 “Selection” view, which aims to gauge the interest level of users in the proposed image.

Fig. 4.2 “Forum” view, which allows users to engage in a discussion about the image, once it has been deemed interesting in the selection scene.
4.3.1 Dialogical Framework

In the EI model of the WeCurate application, we can distinguish three roles:

1. **User Assistant**: the agent that represents the user in the institution.
2. **Media Provider**: the agent that sends the image files to the users.
3. **Manager**: the agent that is responsible for creating the scene instances.

The institution should contain exactly one agent playing the role of Media Provider and exactly one agent playing the role of Manager. Furthermore there is one agent playing the role of User Assistant for each user participating in the application.

4.3.2 Performative Structure

The workflow of WeCurate can be divided in three stages: selecting interesting images, discussing selected images, and voting on disputable images. These three stages can naturally be represented in an EI-specification as three separate scenes. Each of these scenes will have exactly one instance. All user assistants should always be together in the same scene instance. Note that, as illustrated by Figure 4.4, the Media Provider only enters the Selection scene, and this agent stays in that scene...
Fig. 4.4 The performative structure of the WeCurate institution

until the institution closes. The performative structure of the institution is presented by Figure 4.4. Next, we present the three scenes in more detail.

1. Selection: The Selection scene, whose protocol is illustrated by Figure 4.5, is the first scene the users enter when they have entered the WeCurate institution. The protocol starts with the server sending an image to the users. In an Electronic Institution we can represent this by the agent playing the role of Media Provider sending a message to the User agents containing a URL of the image on the server. Note that in the EI-infrastructure it is not possible to send files with messages, so we cannot send the image itself. In the application the users can then express their opinion about the image by moving a slider up or down,
Fig. 4.5 The WeCurate selection protocol

Fig. 4.6 The WeCurate forum protocol

Fig. 4.7 The WeCurate voting protocol
where the slider being down means that the user totally dislikes the image, and the slider being up means he or she totally likes the image. In the Electronic Institution, this is modeled as a User Assistant sending a message to all the other User Assistants containing a number between 0 and 1, that represents the user’s preference level.

Also users can zoom in on parts of the image. Whenever a user zooms in, the other users are able to see on their screens where the first user is zooming. In the institution, this is modeled by means of a ‘zoom’ message sent to all other User Assistants that contains the coordinates of the location where the user is zooming.

Once a user has made his or her decision about the image, he or she can click ‘next’. When the first user clicks ‘next’ a timer starts counting down from 10 to 0. The scene then continues until either all other users have also clicked ‘next’ or till the timer expires. Whenever you have clicked ‘next’ you cannot do anything anymore so you have to wait till the protocol finishes. We have modeled this by introducing two states in which the users can zoom. Whenever the first user sends a ‘next’ message the protocol moves from the first to the second state. The difference between these two states is that the second state contains an arc with a timeout, to represent the countdown.

Next, two things may happen: either the manager decides that the image was not interesting enough for further consideration, in which case a new image will be requested from the Media Provider, or the users move on to the Forum scene for further discussion.

2. Forum: In the forum scene, whose protocol is illustrated by Figure 4.6, the users further ‘discuss’ the image from the previous scene. They can not only express their overall impression of the image, but they can also express which aspects of the image they like or don’t like. In the real application they do this by adding tags to the image and by increasing or decreasing the size of those tags they give more or less weight to those tags. For example if you like an image especially because it has a lot of colours, then you add the tag ‘colours’ and make it very large. In the Electronic Institution, we represent this as a message containing a string and a number, representing the tag itself and its size respectively. This continues until all the users have indicated they want to move on to the vote scene, or till a timer expires.

3. Voting: Finally, when the users are in the voting scene they can vote whether they want the image to be added to their shared photo album or not. In the Electronic Institution, this is represented by a message containing either of the three values: ‘yes’, ‘no’, or ‘abstain’. After voting the users move back to the Selection scene. The protocol of the voting scene is presented by Figure 4.7.
4.4 Integrated Technologies: Automated Decision Making

In WeCurate, the automated decision making of the agents is undertaken by different models: preference aggregation [23], multiple-criteria decision making [1, 17] and voting. Preference aggregation allows having a quick understanding whether the users consider an image as interesting or not. To this end, each user expresses an image preference and a collective decision is achieved by preference aggregation. Multiple-criteria decision making is a collective decision taken as a result of a negotiation protocol. Users exchange image arguments according to an argument-based multiple criteria decision making protocol. Voting takes the majority of the agents’ decision into account. In the following sections we describe these decision models and how they have been adopted in the different scenes of the WeCurate system.

4.4.1 Selection Scene and Preference Aggregation

The main goal of each user running in a selection scene is to express a preference about the image currently browsed. When the scene is finished, the UserAssistant agents compute an evaluation of the image – the image’s interestingness of the group of users – based on preference aggregation in order to decide whether to proceed with a forum scene or to go back to a selection scene (with a different image).

Let $\mathcal{I} = \{im_1, \ldots, im_n\}$ be the set of available images where each $im_j \in \mathcal{I}$ is the identifier of an image. The image preference of a user w.r.t an image is a value that belongs to a finite bipolar scale $\mathcal{S} = \{-1, -0.9, \ldots, 0.9, 1\}$ where $-1$ and $+1$ stand for “reject” and “accept” respectively. Given a group of users $\mathcal{U} = \{u_1, u_2, \ldots, u_n\}$, we denote the image preference of a user $u_i$ w.r.t an image $im_j$ by $r_i(im_j) = v_i$ with $v_i \in \mathcal{S}$.

A preference aggregator operator for merging the preferences of a group of $n$ users $\{u_1, u_2, \ldots, u_n\}$ w.r.t an image $im_j$ is defined as a mapping $f_{agg} : \mathcal{S}^n \rightarrow \mathcal{S}$. Then, a decision criterion for the collective decision about the interestingness of an image $im_j$ can be defined as:

$$\text{int}(im_j) = \begin{cases} 1, & \text{if } 0 < f_{agg}(r) \leq 1 \\ 0, & \text{if } -1 \leq f_{agg}(r) \leq 0 \end{cases}$$ (4.1)

where $r = \{r_1(im_j), \ldots, r_n(im_j)\}$ is a vector consisting of the image preferences of $n$ users w.r.t. an image $im_j$.

In the select scene, $f_{agg}$ and $\text{int}(im_j)$ are instantiated by three different preference aggregators as we will see.
4.4.1.1 Image interestingness based on arithmetic mean

A straightforward way to aggregate image preferences is by arithmetic mean. The image interestingness of a group of users \( \{u_1, u_2, \ldots, u_n\} \) w.r.t. an image \( im_j \) based on arithmetic mean, denoted by \( f(r) \), and its respective decision criterion are defined as:

\[
\overline{f(r)} = \frac{\sum_{i=1}^{n} r_i}{n} \quad (4.2)
\]

\[
\overline{\text{int}}(im_j) = \begin{cases} 1, & \text{if } 0 < \overline{f(r)} \leq 1 \\ 0, & \text{if } -1 \leq \overline{f(r)} \leq 0 \end{cases} \quad (4.3)
\]

Therefore, the system proceeds with a forum scene when \( \overline{\text{int}}(im_j) = 1 \), while the system goes back to a select scene when \( \overline{\text{int}}(im_j) = 0 \).

Since the arithmetic mean cannot take users’ activity into account, we define two preference aggregators which can consider the users’ engagement in the select scene.

4.4.1.2 Image interestingness based on weighted mean

In this case, each UserAssistant agent also stores the zoom activity of its user. The zoom activity is a measure of the user’s interest in a given image and, as such, it can be used to calculate the importance or weight of the image preference of each user. To this end, let us denote the number of image’s zooms of user \( u_i \) w.r.t. an image \( im_j \) as \( z_i(im_j) \). Then, we can define the total number of zooms for an image \( im_j \) as \( z(im_j) = \sum_{i=1}^{n} z_i(im_j) \). Based on \( z(im_j) \) and the \( z_i \)'s associated with each user, we can define a weight for the image preference \( r_i \) of user \( u_i \) as \( w_i = z_i/z(im_j) \).

Therefore, the image interestingness of \( n \) users w.r.t. an image \( im_j \) based on weighted mean, denoted by \( f_{wm}(r) \), and the respective decision criterion can be defined as:

\[
\overline{f_{wm}(r)} = \frac{\sum_{i=1}^{n} r_i w_i}{\sum_{i=1}^{n} w_i} \quad (4.4)
\]

\[
\overline{\text{int}_{wm}}(im_j) = \begin{cases} 1, & \text{if } 0 < \overline{f_{wm}(r)} \leq 1 \\ 0, & \text{if } -1 \leq \overline{f_{wm}(r)} \leq 0 \end{cases} \quad (4.5)
\]

4.4.1.3 Image interestingness based on WOWA operator

An alternative criterion for deciding whether an image is interesting or not can be defined by using a richer average operator such the Weighted Ordered Weighted Average (WOWA) operator [18]. The WOWA operator is an aggregation operator
which allows to combine some values according to two types of weights: i) a weight referring to the importance of a value itself (as in the weighted mean), and ii) an ordering weight referring to the values’ order. Indeed, WOWA generalizes both the weighted average and the ordered weighted average [21]. Formally, WOWA is defined as [18]:

\[ f_{wowa}(r_1, \ldots, r_n) = \sum_{1 \leq i \leq n} \omega_i r_{\sigma(i)} \quad (4.6) \]

where \( \sigma(i) \) is a permutation of \( \{1, \ldots, n\} \) such that \( r_{\sigma(i-1)} \geq r_{\sigma(i)} \) \( \forall i = 2, \ldots, n \), \( \omega_i \) is calculated by means of an increasing monotone function \( w^*(\sum_{j<i} p\sigma(j)) - w^*(\sum_{j<i} p\sigma(j)) \), and \( p_i, w_i \in [0, 1] \) are the weights and the ordering weights associated with the values respectively (with the constraints \( \sum_{1 \leq i \leq n} p_i = 1 \) and \( \sum_{1 \leq i \leq n} w_i = 1 \)).

We use the WOWA operator for deciding whether an image is interesting in the following way. Let us take the weight \( p_i \) for the image preference \( r_i \) of user \( u_i \) as the percentage of zooms made by the user (like above). As far as the ordering weights are concerned, we can decide to give more importance to image preference’s values closer to extreme value such as \(-1 \) and \(+1\), since it is likely that such values can trigger more discussions among the users rather than image preference’ values which are close to 0. Let us denote the sum of the values in \( \mathcal{F}^+ = [0, +0.1, \ldots, +0.9, +1] \) as \( s \). Then, for each image preference \( r_i(im_j) = v_i \) we can define an ordering weight as \( w_i = r_i(im_j)/s \). Please notice how the \( p_i \)’s and \( w_i \)’s defined satisfy the constraints \( \sum_{1 \leq i \leq n} p_i = 1 \) and \( \sum_{1 \leq i \leq n} w_i = 1 \).

Then, a decision criterion for the interestingness of an image \( im_j \) based on \( f_{wowa}(r_1, \ldots, r_n) \) can be defined as:

\[ \text{int}_{wowa}(im_j) = \begin{cases} 1, & \text{if } 0 < f_{wowa}(r) \leq 1 \\ 0, & \text{if } -1 \leq f_{wowa}(r) \leq 0 \end{cases} \quad (4.7) \]

### 4.4.2 Forum Scene and Argument-based Multiple Criteria Decision Making

The main goal of the users in a forum scene is to discuss an image by pointing out what they like or dislike of the image through image arguments based on tags. During the tagging, each user is associated with an overall image preference that is automatically updated. Whilst tagging is the main activity of this scene, users can also engage in bilateral discussions with the aim of reaching mutual agreements about keeping or discarding an image. When a user is tired of tagging, he can propose the other users to move to a vote scene. In this case, an automatic multi-criteria decision is taken in order to decide whether the current image can be added or not to the image collection without a vote being necessary.
In our system each image is described with a finite set of *tags* or *features*. In what follows, we show how weighted tags, that is, tags associated with a value belonging to a bipolar scale, can be used to define arguments in favor or against a given image and to specify a multiple criteria decision making protocol to let a group of users to decide whether to accept or not an image.

### 4.4.2.1 Arguments

The notion of argument is at the heart of several models developed for reasoning about defeasible information (e.g. [10, 15]), decision making (e.g. [4, 7]), practical reasoning (e.g. [5]), and modeling different types of dialogues (e.g. [3, 16]). An argument is a reason for believing a statement, choosing an option, or doing an action. In most existing works on argumentation, an argument is either considered as an abstract entity whose origin and structure are not defined, or it is a logical proof for a statement where the proof is built from a knowledge base.

In our application, *image arguments* are reasons for accepting or rejecting a given image. They are built by users when rating the different tags associated with an image. The set \( \mathcal{T} = \{t_1, \ldots, t_k\} \) contains all the available tags. We assume the availability of a function \( \mathcal{F} : \mathcal{I} \rightarrow 2^{\mathcal{T}} \) that returns the tags associated with a given image. Note that the same tag may be associated with more than one image. A tag which is evaluated positively creates an *argument pro* the image whereas a tag which is rated negatively induces a *argument con* against the image. Image arguments are also associated with a weight which denotes the *strength* of an argument. We assume that the weight \( w \) of an image argument belongs to the finite set \( \mathcal{W} = \{0, 0.1, \ldots, 0.9, 1\} \).

The tuple \( \langle \mathcal{I}, \mathcal{T}, \mathcal{F}, \mathcal{W} \rangle \) will be called a *theory*. An argument is defined as:

**Definition 4.1 (Argument).** Let \( \langle \mathcal{I}, \mathcal{T}, \mathcal{F}, \mathcal{W} \rangle \) be a theory and \( im \in \mathcal{I} \).

- An *argument pro* \( im \) is a pair \((t, v), w, im)\) where \( t \in \mathcal{T}, v \in \mathcal{S} \) and \( v > 0 \).
- An *argument con* \( im \) is a pair \((t, v), w, im)\) where \( t \in \mathcal{T}, v \in \mathcal{S} \) and \( v < 0 \).

The pair \( (t, v) \) is the *support* of the argument, \( w \) is its *strength* and \( im \) is its *conclusion*. The functions \( \text{Tag}, \text{Val}, \text{Str} \) and \( \text{Conc} \) return respectively the tag \( t \) of an argument \((t, v), w, im)\), its value \( v \), its weight \( w \), and the conclusion \( im \).

In our application, we are mainly interested in two things: i) to have a synthetic view of the opinion of a given user w.r.t. an image and ii) to calculate whether the image can be regarded as worthy to be accepted or not. In the first case, we aggregate the image arguments of a user \( u \) to obtain his overall image preference \( r_i^u \). Instead, for deciding whether an image is accepted or rejected by the whole group we define a multiple criteria operator.

**Definition 4.2 (Multiple criteria decision).** Let \( \mathcal{U} = \{u_1, \ldots, u_n\} \) be a set of users, \( im \in \mathcal{I} \) where \( \mathcal{F}(im) = \{t_1, \ldots, t_m\} \). Given the following user’s arguments:
The overall image preference of a user \( u_i \) denoted by \( r_i^\ast (im) \) is defined as:

\[
    r_i^\ast (im) = \frac{\sum_{1 \leq j \leq m} v_{i,j} \cdot w_{i,j}}{\sum_{1 \leq j \leq m} w_{i,j}}
\]  

Then, a multiple criteria decision operator can be defined as:

\[
    \text{MCD}(im) = \begin{cases} 
    1, & \text{if } \forall u_i, 0 \leq r_i^\ast (im) \leq 1 \\
    -1, & \text{if } \forall u_i, -1 \leq r_i^\ast (im) < 0 \\
    0, & \text{otherwise}
    \end{cases}
\]  

(4.9)

Note that the MCD criterion allows three values: 1 (for acceptance), -1 (for rejection) and 0 (for undecidedness). Therefore, an image \( im \) is automatically added to the image collection if it has been unanimously accepted by the users. On the contrary, the image is discarded if it has been unanimously rejected. Finally, if \( \text{MCD}(im) = 0 \), then the system is unable to decide and the final decision is taken by the users in a vote scene.

### 4.4.2.2 Arguing and Users’ agreement

The main goal of two users running in an argue scene is to try to reach an agreement about to “keep” or to “discard” an image \( im \) by exchanging image’s arguments. The argue scene defines a bilateral argumentation protocol.

- Two users tag an image \( im \) by means of image’s tags (like in the forum scene), and they can propose their image’s tags to the other user:
  - While tagging, their overall image preferences are automatically updated;
- A user proposes an image’s tag to the other user who can either accept or reject it:
  - If the user accepts the image’s tag proposed, then their overall image preferences are automatically updated:
    - If an argue agreement is reached, then the argue scene stops,
    - otherwise, the argue scene keeps on;
  - if the user rejects the image’s tag proposed, then the argue scene keeps on;

Both users can also decide to leave the argue scene spontaneously.
Informally, an *argue agreement* is reached when the image preferences of the two users agree towards “keep” or “discard”. Let \( r_i^a(im) \) and \( r_j^a(im) \) be the image’s preferences of user \( u_i \) and \( u_j \) respectively. Then, a decision criterion for deciding whether an argue agreement is reached can be defined as:

\[
\text{argue}(im) = \begin{cases} 
1, & \text{if } (0 \leq r_i^a(im) \leq 1 \land 0 \leq r_j^a(im) \leq 1) \\
\land (-1 \leq r_i^a(im) < 0 \land -1 \leq r_j^a(im) < 0) & \text{otherwise} 
\end{cases}
\]  

Therefore, an argue scene stops when \( \text{argue}(im) = 1 \). Instead, while \( \text{argue}(im) = 0 \), the argue scene keeps on until either \( \text{argue}(im) = 1 \) or the two users decide to stop arguing.

### 4.4.3 Vote Scene

The main goal of the users running in a vote scene is to decide by vote to add or not an image to the image collection. This decision step occurs when the automatic decision process at the end of the forum scene is unable to make a decision.

In a vote scene, each user’s vote can be “yes”, “no”, or “abstain” (in case that no vote is provided). Let \( v_i \in \{+1, 0, -1\} \) be the vote of user \( u_i \) where \( +1 = \text{"yes"}, -1 = \text{"no"}, \) and \( 0 = \text{"abstain"} \) and let \( V = \{v_1, v_2, \ldots, v_n\} \) be the set of votes of the users in a vote scene. Then, a decision criterion for adding an image or not based on vote counting can be defined as:

\[
\text{vote}(im_j) = \begin{cases} 
1, & \text{if } \sum_{1 \leq i \leq n} v_i \geq 0 \\
0, & \text{otherwise} 
\end{cases}
\]  

Therefore, an image \( im_j \) is added to the image collection if the number of “yes” is greater or equals than the number of “no”. In the above criterion, a neutral situation is considered as a positive vote. \(^1\)

### References


\(^1\) This assumption is made to avoid an undecided outcome in this decision step.


