Description of mail visualization scheme

For my mail visualization, I've envisioned the use of Cartesian space to represent salient patterns in e-mail (which can just as easily be extended to newsgroup or message board posts). A problem I've encountered in dealing with mass amounts of email is filtering through unimportant or non-urgent messages, thereby limiting the time I have to address essential issues via e-mail. Thus, my visualization is centered on making more visible those threads that are urgent or require my immediate attention.

For that reason, the scope of my visualization is designed to serve an individual user or class of users with similar definitions and keywords of importance, rather than as a dashboard for scrutiny by a third party, scientific analysis, etc.

I decided to use Cartesian space because objects placed on it captures four dimensions easily. On a plane, we can place objects according to two parameters and see how they compare to other objects on the plane - observe Randall Munroe's fruit chart (http://xkcd.com/388/) wherein he ranks fruit along axes of tastiness and ease of use. One could easily represent another quality, such as price, by altering a quality of the object on the plane - such as making pomegranates large and bananas small to reflect their respective prices at the grocery. One could easily represent other qualities in this way, *ad infinitum*, however at the cost of readability and legibility. A practical limit in representing parameters in a plane is thus three. By extension, one can use space to represent four characteristics. And due to the relatively unbounded computer power at our fingertips these days, it is possible to process an manipulate data in this way.

A base case is as follows: If I receive a message and nothing else, it will show up as a rectangular prism with some determinate amount of volume. On the x-axis is time - if a thread (or single message) is newest, it will be furthest from the origin. On the z-axis is message count. In the base case, it would be a single cube that the user can click on and access its underlying message. If another message on that thread were sent, another cube would stack on top. Reference my wonderful Paint drawings below:
As you can see, the cubes are distinguished from each other by color. This can easily be replaced by other visual indicators, such as an avatar of the person who sent it.

Now let's see what happens when I get another message on a completely different thread.

In this manner, the space gets populated like a newsfeed, the newest items appear "closer" to the user and are brought to the user's foremost attention.

Now where does importance come in? For the purposes of this assignment, let's assume I've written an algorithm that assigns an importance value on messages containing set
keywords of my choosing, like "important," "urgent", "90% off", etc. and measures the emotional state of the author by his or her tone, relative grammatical correctness, etc. The length of the message object gets extended in proportion to this importance value. Observe:

Figure 4. Four messages in three threads, one message with higher importance

Due to the larger size of the message object, the user would be able to distinguish it more easily even if newer messages are incoming or if an existing thread receives a new message:

Figure 5. Five messages on three threads - an older thread comes to the front when a newer message gets added - and the importance is judged higher than the original message.

The user would not be limited to this view and could zoom, rotate, scroll through, and manipulate his or her view of the visualization at will. For example, if the user wanted to see lengthy threads containing messages of high importance, he or she would rotate the view to see the YZ-plane:
In summary, this visualization gives the user a view into the importance, length, relative timing, and owners of different e-mail threads and be able to better sort through the clutter of messages based on personal priorities, desire to read messages by a certain user, checkup on recurring threads, etc.