

## Scoring Vectors:

Instead of a simple 'up / down' scoring scalar, use a multi-variable scoring vector, which can give descriptive scores instead of just 'good vs bad'. Examples could be:

Art vs Science – Funny vs Serious – Novice vs Expert – Rude vs Polite - ... - as you need- ...

The actual labels used would be chosen by whoever is implementing the system. By using a different address list (ie. 'The List' from previous slide), you could implement any number of different 'Cloud Comment' networks, which could compete using different 'value systems'.

When the comments are searched, or queued into a feed, a user's 'Feed Template' affects the sorting of the comments, by stating which scalars to sort by, and which algorithm to use.

## 'Bossiness rating':

If a user exceeds their 'scoring quota' (per hour, say), then their scores are exponentially reduced - by of a factor of 2, then 4, 8, 16 etc – until their impact becomes negligible. Once a user stops excessively scoring things, this factor goes back to 1. ('The more you shout the less they listen.')

This could be done without keeping detailed voting records for each user, and would also allow voting for some things more than once, at the expense of being unable to vote for other things.

You get a 'quota of votes' to use in a time period. In other words, a 'voting rate', instead of a 'voting allowance'.

## 'Lazyness' rating:

If people don't score anything at all, then the comment system won't function properly, so elements of a user's score (the ones that affect their position in the 'feed queues'), get reduced to a base level, and increase as their scoring rates increase.

However, if they score things excessively, then their 'bossiness rating' goes up.