CSCW - Computer Supported Cooperative Work

In order to make a CSCW application work , all the users need to be incentivised to contribute. For that to happen , they need to be able to see some value in the application. Traditionally , most CSCW applications fail because there is a disparity between who does the additional work and who gets the benefit.

Example : Automatic Meeting Scheduling :

- Requirement : Everyone in the organisation maintains digital calendars.
- Reality : Only the top management officials maintain digital calendars. The lower level workers do not see a need to.
- This application adds value to the person calling the meeting - usually a CXO level exec but does not add any value to other workers. It in fact creates the additional burden of having to maintain a digital calendar.
- **This disparity between who does the work and who benefits is the major reason why most CSCW applications fail.**

This paper identifies three different reasons for the failure :

1. **Disparity between who does the work and who benefits :**
    - Should you leave the choice to individual discretion ? NOOO - communication breaks down in that case and a CSCW app without communication is a misnomer.
    - Should you use coercion ? NOOO - ethical issues aside , you can't keep doing it to every new CSCW app you introduce , costs will far outweigh the benefits. A systemic change in roles is acceptable if there is change at the system level and not for application level changes.
    - What should I do instead ?
        - Build additional features for every set of users and not only for the managers
        - Reduce the extra work or at least incentivise the extra work being done.
        - This is hard because unlike in a single user application , you can't settle for a lowest common denominator , you have to factor in all cohorts of users.
2. **No Intuitive Decision Making :**
    - Managers who make the decisions of shipping changes use their intuition. Intuition developed through experience with single user applications. This intuition does not make sense for a multi user CSCW app.
    - Also , our intuitions are pretty good at assessing what people like us would prefer. Not very good for assessing what everyone would prefer. You need to research to understand that.
3. **Failure to learn from Experience :**

- This is because it is intrinsically much difficult to evaluate CSCW apps.
- It is easier to set up a usability study of a single user app in a lab setting over an hour or two. It is not the case for a CSCW app. You cannot simulate group interactions while also accounting for the differences in the economic , social , educational backgrounds of all the sets of users we anticipate would use the app.
- Group interactions are spread out over a longer span of time ~ 2-3 days

Case Studies :

1. **Digitised Voice Annotation :**
    - **Advantages :**
        - Speaking is comfortable and faster than typing.
        - Speech conveys nuance and emotion enabling you to be more expressive.
        - Speech annotation reduces visual clutter.
    - **Disadvantages :**
        - Reading is faster than listening.
        - Reviewing voice is harder than reviewing text.
    - In this case , it can be seen that the speaker gets all the benefits while the listener has to do the additional work.
    - This will work in a setting where all the users have the equal likelihood of being speakers and listeners at different times.
2. **Project Management :**
    - A project management application running on a distributed system is the demonstration of ultimate potential of CSCW - My Bachelor's Thesis does the same for construction management (about time I write a concise 10 page summary of my thesis . No one wants to read a 100 page thesis. Sigh !!)
    - **Advantages :**
        - scheduling and chronicling of activities
        - creation and evaluation of plans
        - monitoring of resources and applications
    - **Disadvantages :**
        - Clearly , the benefits are enjoyed by the project manager whereas the arduous task of keeping the information current is left to the rest of the users.
    - Coercion is an option but a better option would be provide benefits to the users to keep the information current. I believe I've done this in my BTP. Please checkout.
    - Another challenge is to build intuitive interfaces for information input.
3. **Natural Language Interface to Shared Databases :**
    - Useful for the casual or novice users e.g. executives , sales managers etc.
    - NLI will help solve the issue of different people using a CSCW application to different extents.
4. **Group Decision Support :**
    - Again , useful for the decision maker , not so much for the rest of the users.

    - Very similar to the project management example.

Summary :

- Multi User Applications are different from Multi User Systems.
- Multi User Applications are different from Single User Applications.