Cloudy Forecast: An Exploration of the Factors Underlying Shared Repository Use

Charlotte Massey
University of California at
Santa Cruz,
High St, Santa Cruz, 95064.
cmassey@ucsc.edu

Thomas Lennig
Imimtek
Palo Alto, 94303.
thomas@imimtek.com

Steve Whittaker
University of California at
Santa Cruz,
High St, Santa Cruz, 95064.
swhittak@ucsc.edu

ABSTRACT
Many teams are now adopting *shared repositories* for their work. Such adoption is paradoxical, however, as past research has repeatedly shown major *co-organizational barriers*; teams cannot agree a common organizational scheme, making it difficult to retrieve information organized by others. Another barrier is *email competition*; email provides a reliable alternative for distributing files that are then personally organized. To address this paradox, we explored how 27 participants actively using shared repositories overcome these barriers in a qualitative study. We found teams addressed co-organization using 4 strategies. First they create *ContentMaps* that provide explicit structure to organize shared information. Participants also co-organize using implicit strategies based on *task structure*, *expertise*, and *tool affordances*. Greater shared repository use also leads to a changed role for email. Versioning problems mean email is not used for distributing attachments, instead for task management. We present technical implications suggesting how new tools might be better integrated with email facilitating these continued email uses.

Author Keywords
Shared repositories; co-organization; email competition; ContentMaps; versioning.

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION
People who collaborate usually generate shared digital content. But collaborators typically face a dilemma about how to *share* the information they create together. They need to choose between storing shared files in a *shared repository* (e.g. GoogleDrive) or sharing *email attachments* and then organizing them in personal repositories.

There are excellent intuitive arguments for using shared

repositories rather than email. Relying on email involves all collaborators independently managing their own personal collection of relevant files, duplicating time and cognitive effort. Furthermore, email distribution leads to well-documented problems in retrieving and reconciling *different versions* of the same file [7,8,30]. Many organizations therefore have a policy of encouraging teams to use a common repository when sharing files [20].

Despite these arguments, the history of shared repositories has been one of repeated failure. The first such repositories were developed over 20 years ago [11,19]. They were unsuccessful initially and subsequent research continues to document outstanding problems. One commonly observed barrier to adoption is *co-organizational failure* when participants *cannot agree a shared organization* [11, 19, 24, 27]. Many studies describe disagreements arising from different organizational strategies among members of the collaborative team. One consequence of co-organizational failure is balkanization where individuals store information in their own private region of the common repository, undermining its function as a shared resource, and making it difficult for others to find that information [24].

A second problem with shared repositories is *email competition*. Successful collaboration depends on people being aware of coworker actions, allowing them to co-ordinate interdependent tasks. To avoid destructive co-edits and costly version reconciliation, interdependent tasks like collaborative document editing require that collaborators know which is the most recent draft, who is editing it, and when a draft is completed. One important limitation of early shared repositories was that participants had to *explicitly access* the repository to discover whether new versions had been created [11]. To avoid this, participants sent emails alerting others about version changes, sometimes including new files as attachments. Such attachments undermined the shared repository leading some collaborators to revert to email for sharing [20, 27].

New shared repositories like Dropbox and GoogleDrive provide potential solutions to versioning with features like alerting and concurrent editing. Nevertheless they have not overcome all outstanding problems. Indeed recent studies of modern repositories have identified additional problems, including *learning new conceptual models* [22].

Furthermore the proliferation of different shared repositories now means collaborators not only have to agree how to organize information *within* each tool, but also how information is to be distributed across multiple *different* tools [22, 27]. People now have to decide whether that document is in Drive, Dropbox, or on the shared fileserver.

Despite these problems, we have recently seen greater adoption of shared repositories. Dropbox reports 100 million active users who upload a billion files each day, and Drive reports 10 million users, although these numbers include private usage [1, 2]. This widespread adoption gives rise to a *paradox*. If shared repositories are being extensively deployed, how are users overcoming the many barriers documented in prior work?

One possible answer may be the use of *keyword search*, which potentially finesses the problems of co-organization. Keyword search may allow collaborators to overcome *co-organization problems*, supporting successful retrieval even when searchers don't know *exactly where* the information is stored [12, 25].

We address the *adoption paradox* for shared repositories, to determine how and why users are overcoming well-documented problems. We interview 27 people to determine success factors and characteristics of effective deployment. With greater uptake of these tools, we ask:

- What new factors promote successful adoption: *how* are users of new tools such as Drive and Dropbox *overcoming the co-organizational barriers* of older tools?
- How *successful are users at retrieving information* given that others may have organized this information? Do users rely on keyword search to avoid having to navigate others' organizational systems?
- What is the *relationship between shared repositories and email*? Are these in competition, or are they used synergistically?

We address *informal collaboration* rather than more formal processes, e.g. software engineering where teams often deploy dedicated versioning tools to handle collaborative problems. Another important characteristic of our study is that we examine shared repositories in *everyday work contexts*. Our participants talked about familiar shared files accessed on a daily basis and not long-term archives. This focus contrasts with much prior work that examined shared repositories in *academic research contexts*, where there are different demands for systematic organization to guarantee long-term access and retrieval [31].

RELATED WORK

Co-organizational Difficulties Within and Across Tools

Many studies have documented problems collaborators experience in agreeing a common organizational scheme for shared repositories [11, 19, 24]. Early work enumerated different personal organizational strategies (scruffies vs neats, savers vs purists...), and the problems that ensued when these incompatible personal schemes were applied to a shared research repository.

Personal Information Management (PIM) research examines how *individuals* organize information. One repeated PIM observation is large individual differences in how people organize information to promote later retrieval [10, 31]. People differ in how much information they keep, in the extent to which they organize, and in the different methods used to organize their information [9, 17, 31]. Similar individual differences in organization occur with physical files [21], email [7, 13] and web resources [3, 5, 17]. These organizational choices affect retrieval: people with shallow broad file hierarchies navigate to information faster than those with deeper, narrower structures [10].

Participants in *shared* repositories may therefore be motivated to use personal organization because they worry they will be unable to find information organized using someone else's framework [24]. [19] argues that "People adding and retrieving information in group information systems must mash their often idiosyncratic categories, indices, schema and information routines" (p. 243).

These problems are exacerbated with newer social media and cloud based tools [26, 27]. [22] describes how collaborators have incompatible and often incorrect models of how shared tools work, undermining their utility. Learning a new system represents a significant adoption barrier. If a single collaborator fails to learn the new shared repository this undermines its collaborative utility potentially leading participants to abandon it [16]. Other problems arise when individuals attempt to stratify their use of multiple tools according to different online identities or roles (e.g. work vs personal). Collaborators may be unaware of these personal decisions leading them to post information in unexpected places [27].

We explore the implications for shared repositories of these findings. How do people organize and access shared information when successful access to personal information often depends on exploiting idiosyncratic organizational schemes? We examine whether and how active users of new shared repositories are able to reconcile personal and collaborative organization.

Keyword Search, Organization and Retrieval

Keyword search has potential implications for shared repositories. If search is successful, users no longer have to navigate through file hierarchies to retrieve information, as they can search for it *without knowing exactly where it is* [12, 25]. Search is provided for Drive, and regular desktop search accesses the contents of shared repositories like Dropbox.

Although keyword search has not been widely studied in shared repositories, there is extensive work examining its use in *personal* repositories. Somewhat surprisingly, navigation through the file system is the preferred retrieval strategy [9]. Search is only used as a 'last resort' when people are unsure where information is located [9]. In this study we address whether search solves co-organization

problems in shared repositories, allowing retrieval without knowing where information is stored.

Distribution, Versioning, and Co-Editing

Email has long been used as a tool for distributing shared materials [8, 30]. Its benefits for sharing include that it is simple, highly familiar and participants do not have to install or learn a new application. It also finesses the co-organizational problem because each participant organizes materials independently, allowing them to each organize information as they choose. However there are problems associated with using email in this way. Email is difficult to manage efficiently, especially when used for collaborative tasks [8, 30]. If multiple versions of a file are circulated in email, overloaded inboxes mean that collaborators may accidentally access and edit outdated versions. This in turn leads to complex reconciliation of different versions. While newer tools support collaborative task management in email [7,18], these remain research prototypes without widespread adoption.

In shared repositories, two main approaches address problems of versioning. The first allows concurrent edits on a single version, as in Drive or Sharepoint. In the second, peripheral alerts inform users when changes are made to a shared repository (e.g. in Dropbox). If two participants accidentally co-edit then different versions are saved leaving users to reconcile these. Even with shared repositories, email still plays an important role in task management. When participants use a shared repository, they often use email to describe new versions and to allocate workflow [8, 14]. This study explores how participants address versioning both in email and in shared repositories.

Organizational and Transactive Memory

Collaborative information management can be viewed as a form of *organizational memory* [4, 6]. Furthermore, *transactive memory* [28] describes how in collaborative settings, people delegate remembering to familiar others, who they know have superior recall in that domain. Thus one member of a couple might excel at remembering family birthdays, whereas the other might best remember passwords or bank account details. By playing to their strengths, they out-remember what they can achieve alone. We later explore whether transactive memory concepts apply to participants' attempts to co-organize shared files, with organizational tasks being delegated to familiar others with known areas of expertise.

METHOD

Participants and Their Work Settings

We interviewed 27 participants of whom 9 were females. Their ages ranged from 24-52. Our intention was to explore *successful usage strategies*, so we targeted people who incorporated shared repositories into day to day work tasks.

They were recruited using convenience sampling in the San Francisco Bay Area and worked for multiple different organizations, with heterogeneous job types. See Table 1. None of the participants was an academic, a decision motivated by prior research suggesting that academics have rather specific strategies for managing personal information [31]. Companies ranged in size from individual consultants and small 3 person startups to massive multinationals.

Interviews

Two interviewers conducted semi-structured interviews that explored the following topics:

- *Co-organization*: how shared repositories were co-organized, who was responsible for their organization, as well as collective problems and or disagreements associated with creating the organization,

- *Retrieval*: how participants retrieved information from repositories, in particular whether participants relied on search, as this has implications for needing to organize,

- *Email*: the relationship between shared repositories and email, and perceived strengths and weaknesses of each.

Interviews were based around *collaborations*: we first asked participants to describe 3-5 current collaborations. Participants then talked about shared repositories used for each collaboration, including: the data kept in each tool, who creates that data, how the tool is organized, who organizes it and their finding strategies within the tool. We then explored the relationship between each shared repository and email.

Interviews were conducted over Skype using remote desktop sharing. There was one exception, (P12), who we interviewed face to face. Participants showed us each shared repository, illustrating how it was organized and used, and demonstrating various retrieval strategies used with that tool. We then probed participants' collaborative email practices, asking them to compare sharing using email attachments with shared repository use. Interviews lasted 45-60 minutes: they were audio recorded and transcribed. Participants typically talked about 1-3 different collaborations in depth, but also provided an overview of the entire set of shared repositories they used.

Data Analysis and Coding

Analysis involved four phases: identifying initial themes, reaching consensus, applying themes and reliability checks. Initial coding schemes were developed by two coders who independently analyzed 7 interviews, identifying themes, based on categories suggested by prior research [11, 22, 24, 27], and developing new themes. These two coders then reconciled their separate coding schemes, agreeing on key themes. One of the initial coders then coded the remaining interviews coding using the agreed themes. Finally two additional independent coders checked each interview to ensure reliability of coding. This process identified:

Table 1: Participants (N=27)

Total Participants	Participant Numbers	Job Type
2	P23, P24	CEO
1	P6	Startup Cofounder
1	P17	VP Marketing
1	P15	Sales Director
1	P9	Marketing Coordinator Project Manager
5	P1, P2, P5, P7, P14	Researcher - Manager
1	P3	Researcher
1	P11	Lead Software Engineer
3	P10, P19, P21	Software Engineer
1	P27	Presales Engineer
1	P16	QA Engineer
1	P8	Mechanical Engineer/Test Manager
1	P20	Junior Game Designer
1	P4	Junior Technical Writer
1	P26	IT Director
1	P13	Technology Integrator
1	P12	Technical Support
2	P18, P22	Teacher
1	P25	Contract Photographer

problems and solutions to co-organization, retrieval strategies, and email comparisons.

RESULTS

Teams address co-organization using 4 strategies. They created *ContentMaps* providing explicit structure to organize shared information. Participants also co-organized using implicit strategies based on *task structure*, *expertise*, and *tool affordances*. Greater shared repository use also led to a changed role for email.

Individuals Use Multiple Repositories

Consistent with other recent studies [22, 27] most participants used *multiple* shared repositories (e.g. Dropbox and Drive or a wiki and SharePoint). After we had eliminated a small number of repositories used for *personal* storage (e.g. personal backup or access from multiple devices), the mean number of shared repositories used by each participant was 2.48, with a minimum of one and a maximum of 5. Table 1 shows Dropbox and Drive were most frequently used, and most participants used both. People using just one repository used Dropbox or Drive. About a third of participants used shared network folders (i.e. intranet folders on an organizational server). Wikis and Sharepoint were less common, and there were a small set of tools used by just a single participant.

Table 2: Frequency of Use of Collaborative Tools (N=68)

Examples	Frequency
Dropbox	23
GoogleDrive: Gdocs, Gspreadsheet and Gslides	23
Shared Network Folders (Fileshare, Filezilla)	10
Wikis (Confluence, Communities)	5
SharePoint	4
Box	2
PRIM (in-house repository)	1

Reduced Attachments and Little Use of Search Imply Participants are Co-organizing

The active use of shared repositories implies participants are at least partially overcoming problems described in prior work. In addition participants reported reduced use of email for sharing attachments, which also suggests that shared repositories were becoming more successful. *Every single* participant acknowledged that they were less dependent on email for distributing attached versions.

(p11) E: Do people send you attachments in email?

P: If it's something basic, yes. But the majority of the time it's just a link to the drive or a link to Dropbox.

(p23) E: Would you say that email is primarily for communication or file sharing?

P: Communication. Like 95%. Because there are so many other ways to share documents so Dropbox, GoogleDrive.

A few participants even described frustration when they were forced to revert to using email attachments:

(p17) my [colleagues] will not send attachments; we are dead opposed to it. So if somebody is sending an attachment it's somebody that's not up to speed in that modern stuff.

(p24) [I don't use email] unless I'm forced to because the person that I'm collaborating with is a dinosaur. I don't email file attachments.

However actively using shared repositories and reduced attachment use may not mean that participants are successfully co-organizing. An alternative explanation is that participants are finessing co-organization by relying on *keyword search* for retrieval. Our data suggests that this is not the case; we asked each participant the most common way they retrieved information from shared repositories. Keyword search was used in 29% of retrieval cases:

(p3) I suppose if I really can't find something then I would search. But usually I find it faster to navigate than to search.

(p13) If need be, I'd use the search bar, but it's pretty well organized and pretty, with this at least I'm pretty anal about my folders, so I never really have trouble looking for stuff.

These results indicate that participants do not rely on search to avoid the need to co-organize. Prior work indicates people use desktop search as a 'last resort' when they don't know where information is located. The low search percentage reported indicates that participants feel they know where information is located. The use of keyword search alone cannot account for increased use of repositories, so participants seemed to be achieving co-organization.

Overcoming the Co-Organization Barrier: Explicit Strategies to Organize Shared Materials

Despite being explicitly asked, participants did not describe complex discussions to agree common organizational strategies for shared repositories. They also reported few difficulties in using shared repositories. How then were

participants able to co-organize well enough to encourage tool adoption? Participants described *explicit* and *implicit strategies* used to collaboratively organize. They discussed 4 strategies to create shared structures allowing people to navigate to relevant collaboration materials. The first explicit strategy relies on *ContentMaps,* a term we use for a *structured description* of where those materials are located. There were also 3 implicit strategies relying on *Tasks, Expertise* and *Tools.*

ContentMaps: Explicit Information Structuring by Proactive Leadership

One way to address co-organization was for one person to generate a *ContentMap*, that provided *direct access* to collaboration materials via active links to the shared repository. Fourteen people described using ContentMaps to structure and share information. This could involve a *combination* of different collaborative tools. The most common instances of ContentMaps involved a 'Map' generated in a wiki, task management application or Googledoc. This Map also served as 'front end', pointing directly to underlying data in the shared repository. Maps contained a description with a set of active links to a shared repository such as Dropbox or Box, or links to other Googledocs. In other instances, structuring information was provided in a workflow application such as a task manager, (e.g. Jira, Asana). Of course ContentMaps do not remove the need for information structuring, but such organization is imposed by *the person organizing the tasks*, rather than resulting from a negotiated team discussion.

ContentMaps served to facilitate access to the contents of the repository. If active links weren't provided, the wiki, Googledoc or task manager would instead include an explanation of where key information was to be found and specify which new materials needed uploading to the repository. This description helped other collaborators ensure that they accessed the right versions of information, incorporating updated material into the appropriate places. Maps might also include who to contact about specific information, and occasionally a history of prior interactions. The map is updated as the repository grows in size.

Here a participant describes how his team's work is distributed via an email that contains a ContentMap ('the Tech List') consisting of a Googledoc with a series of links to different jobs for members of his team. Clicking on a link directly locates the relevant document in Drive, allowing him to edit it in place. He does not have the overhead of navigating the underlying structure of the collaborative archive; the materials he needs are accessed directly via links, without having to know how those documents are organized. Here he describes the format of the ContentMap, how it is shared and how linking removes the need for him to understand the repository structure.

(p16) Our QA manager usually creates the Tech Lists 90% of the time, and she'll just share it with us. She'll send an email with tasks, then we'll go to that master list, which is a Googledoc and

we'll find what we need. We just open a link to a document that's stored on Drive straight from the email, it's like seamless and we'd just click on it goes straight into the doc, straight into editing it, so I don't even use that "My Drive," I hardly ever use that.

ContentMaps might also directly reference materials stored in Dropbox, ensuring that updates are carried out on the relevant version of the document:

(p4) you create a directory link inside of the document. And other people can just download them directly and import it over and copy it exactly. So you can put a pointer into the document itself and it'll reference it to this file, even if it is in Dropbox.

Using ContentMaps addresses some versioning problems experienced with email attachments. This participant uses Drive as a ContentMap for shared Dropbox files.

(p23) Our GoogleDrive will reference certain files. Particularly [XY] documents since they're all up on Dropbox so we'll have a Google Drive that says this *particular file in* this *folder on Dropbox needs this done to it.*

Using direct linking to reference materials avoids versioning problems, because the Map points to the exact document that needs to be modified via a shared editor.

(p15) we'll use links to GoogleDrive documents as opposed to full blown attachments. Because it helps us with version control.

The above examples involve combining *two tools*, one as the front end Map, and the other as the shared repository. A related method for successfully accessing shared documents without explicit co-organization involved creating a *single structured shared application* that both provided structure and allowed data entry. Participants here didn't combine multiple tools, but instead structured a *single* shared tool like a Google spreadsheet or document, so that others could contribute concurrently to it. The application was carefully organized to structure collaboration *within* the application itself. Tabs or fields in the shared document were clearly demarcated with task, worker or client names allowing collaborators to concurrently enter data. This allowed participants to simultaneously track complex responsibilities, enter data or post links. Shared applications therefore structured concurrent work, again without the overhead of managing multiple versions that follow from complex email interactions involving attachments. This marketing manager describes how his team appropriated and structured a Googledoc allowing them to subdivide, track and concurrently collate different client leads.

(p15) it's really important to us that we're not redundant [when following client leads]. And so we have this shared document which shows what date did we contact people, who is the rep, and our result. We kinda improvise the functionality with things like GoogleDrive. So you can see I hit up, you know, 18 people I wanted to meet up with last night. We color code it for [clients] that might be interested and we can either eliminate bad candidates or keep up to date with the state of our conversations with our good candidates using GoogleDrive.

Implicit Co-organization: Expertise, Tasks and Tools

However on other occasions multiple team members did co-organize common repositories, with different team members organizing different subparts. As we noted above, this was not done following complex discussions about organizational semantics, nor did participants describe problems with the resulting organization. How then was this co-organization achieved and how did participants know how to organize and find information within it?

Participants used 3 implicit co-organization strategies based around tasks, expertise and tools. Although these strategies seem somewhat obvious, they were effective to a large extent because participants shared *common knowledge* [4, 28] about shared tasks and information. Participants were intimately familiar with their daily work tasks, how these were organized, who was responsible to each subtask and what tools were generally used to carry these out. In contrast much prior work examined academic contexts where tasks are more open-ended, leading to disagreement about how research resources might be categorized. Instead our participants shared common knowledge [28] about their expertise and tasks, which they exploited to determine how information should be organized. They therefore deliberately delegated organization of specific areas of the shared repository to experts, rather than everyone contributing to the entire organizational scheme. Of course there were limits to this form of organization, participants working in large multinationals lacked knowledge about expertise within a large team. However in these contexts participants were usually not actively collaborating with a huge extended team, but instead publishing final versions of documents for others to consume.

Co-organization by Expertise: One very common implicit way to structure shared repositories was around *expertise* (reported by 11 participants). Although materials were shared, individual collaborators took responsibility for parts of the repository related to *personal job responsibilities or expertise*. People were aware of who had which expertise, and delegated organization to experts. This allowed materials to be exploited by all, without individuals contesting how different subareas ought to be organized. In unusual cases where responsibility was unclear then emails might be exchanged to determine where specific materials were located. The following participant describes his team's distributed expertise, stressing how each person controls different projects (Business/Marketing, Backend, FrontEnd). This shared knowledge about responsibilities makes the resulting organization 'very simple'.

(p6) we are each of us working on a specific problem area. I am working on Business and Marketing and my partner is working on [system] Backend and the other guy is working on the Frontend. When it comes to projects for Customers, there is a project file for all the projects we do. This represents the projects we do for our clients. We know the type of project and once we know from then on [the organization] is very simple.

Co-organization by High Level Task Structure: For predictable workflows e.g. documentation or client management, co-organization reflected *task structure*. Collaborators were each familiar with this structure, allowing them to anticipate where information should be placed to promote easy access for others. 16 participants described using this method to organize shared files. This participant talks about how shared materials are assembled around a simple high level structure when writing a book. Information is architected into a modular structure of chapters, graphics, and style guidelines. Note also that the resulting structure is deliberately designed to be shallow and 'flat' so collaborators don't have to learn a complex hierarchical system.

(p17) We try to be as flat as possible; the root directory is a folder for every project. The only live project is the XX book. And within the XX book there is a folder for chapters, folder for graphics which would be rich media, a folder for my YY press guidelines. it's a high level architecture and so you have logical groupings of content. As soon as the logical grouping you've exhausted the apparent top level topic then everything should be flat underneath.

Often one collaborator (a manager or domain expert) created high-level categories that fit the tasks that the team was carrying out. These categories were later individually populated by different members of the team who contributed new materials. This form of organization was described by multiple participants as 'organic', implying it was both natural and self-organizing. Notice again that this organization mirrors the expertise within the team:

(p15) we have a number of different folders in (Dropbox): Development, investors, competitors, marketing operations, photos, sales, it's loosely grouped into areas where different types of people have different relevancy. Like development, all the coders can go there, investors obviously, it's where you're going to be dealing with people who are going to capitalize the company. Competitors is a very sparse folder with pretty stark and limited look at who our competitors are, we don't have that many. ... this way if we want to go in to look at sales related things for example, it's pretty easy to go to that folder.

One important limitation to these implicit strategies was that organizational difficulty increased with repository size. This participant notes that the success of using simple strategies relies on maintaining only active files:

(p24) When you first start using it it's really straightforward and you only have five or six folders, but after you've used it for a couple of years it's just like any other repository. It does have to be managed, and you do have to have some sort of information architecture with how you're storing things, and you do have to maintain some control or governance over depreciation and deletion of content.

Co-organization by Tool Affordances: Similar implicit solutions were exploited to determine tool choice. Many collaborations used *multiple* repositories, so how did participants agree which of these repositories would contain particular information? Again we saw a reliance on implicit strategies. Different tools had clearly different affordances

for collaborative tasks, allowing participants to *infer* which tool would contain that information. 12 participants described using this method.

(p5) I know that participant data is going to be on [a proprietary shared repository]. I know that things that require a visual representation are going to be in Googledocs and often I remember creating specifically that file or editing it. And everything else is going to be in Dropbox.

Participants mapped different collaborative tasks onto the perceived strengths of each shared tool, allowing them to infer where information would be. Participants expected low fidelity documents to be in GoogleDocs as this allows concurrent editing, but has unsophisticated formatting. In contrast external documents e.g. business plans, contracts or drawings, require professional formatting and access to dedicated applications. These were more likely to reside in DropBox, Shared Filesystems or Sharepoint, as these repositories allowed participants to use dedicated applications to generate higher production quality. Collaborators were aware of tool affordances, and they expected information would be located in the relevant tool.

(p6) in general we work with day to day documents we use Google Docs, for more advanced document sharing think about drawings, physical drawings, or any technical documents or official documents and presentations for work we use Dropbox.

Overall our participants were successful in incorporating shared repositories into their work tasks despite using relatively simple solutions to co-organize. This contrasts with previously documented difficulties. The difference could be explained by restating that these strategies rely on shared team knowledge about tasks, roles, and tools that allow co-organization to be accomplished in a typical work environment. Prior work has noted that in the academic contexts that have often been studied previously, roles, tasks and responsibilities are often far less clear [27].

Finding Materials In A Shared Repository is Usually Unproblematic

Our next question concerned *finding*. We have documented strategies for co-organizing shared repositories. However a critical question concerns how *effective* these strategies are in supporting retrieval. Prior work documents problems in retrieving information from shared repositories, in part because people do not understand where others have placed information [11, 24]. However, our participants did not often report such retrieval problems, even though we specifically asked them about this. Why was this? On some occasions, ContentMaps provided *direct links* to where information was located, removing the need to understand exactly how information was co-organized:

(p10) Usually what you have is a very specific file path that's in the Googledoc. So you don't really search for it or don't really browse for it, you just like put in a direct path and you go straight to it

Even when direct links weren't provided, the relatively simple organizational schemes we described above made collaborative access straightforward. As a result, participants described few problems in retrieving files. Organization reflected *current* work tasks with structures relating to that task or specific people: shared files were familiar and in frequently visited locations. A common retrieval strategy was therefore simply to *browse* common repositories like Dropbox. 17/18 participants who answered how they found items in Dropbox relied on simple navigation. This reliance on navigation suggests that participants knew where information was; as discussed above, prior work suggests people are more dependent on desktop or keyword search if they don't know where information is located [9]. Several participants described how navigation with the shared repository was straightforward, with relatively little use of complex subfolder structures:

(p17) I go direct to files. [Project X] is flat so there is only one subfolder and the reason there is a subfolder code, but other than that it's purely flat. So when I'm trying to find a document I will usually go by date modified, I know a range of date modified and I will know the exact name of the document that I want.

When a ContentMap is used in association with the shared repository, participants exploited explicit structure it provided to assist finding. This participant describes how a task management tool is used to index to underlying files:

(p9) I can navigate by which project, projects are kind of like folders and then there are different tags that we use.... you can see the tag shows me which folder it is in

We asked participants how they found information within each shared repository. Participants were slightly more likely to use keyword search in Drive than other repositories with 6/17 participants describing their finding behavior by saying that they 'usually' searched within Drive. But even in Drive, the majority of participants navigated to content, implying that they understood where information was located.

Overall participants are successful in finding. They exploit ContentMaps or locate familiar sets of files that are organized by tasks and expertise. The three participants who did experience occasional finding problems tended to be retrieving under rather unusual circumstances. They were looking for information that had been created several months previously by an independent team. However attempting to find such old, unfamiliar information was the exception rather than the rule. In most circumstances, for everyday information needs, ContentMaps and work-related implicit structure make access successful.

Back To The Future: Reduced Attachments But Email Still Critical For Alerts and Task Management

One possible implication is that successful practices around collaborative repositories, *reduce participants' dependence on email*. As we noted above, every single participant acknowledged less dependence on email for distributing

attached versions. Although some still executed simple ad hoc collaborations by exchanging email attachments, for most this was not habitual for teamwork. Despite a reduced exchange of attachments, participants were still reliant on email in their collaborations. However they now used email for *different aspects of collaboration*: alerting, task management, commenting/discussion and social finding. This continued interdependence between collaborative tools and email is ironic given that a major initial motivation of collaborative tools was to reduce overall email traffic.

Alerting about Changes in the Shared Repository
A thorny, unresolved problem is how participants are *alerted* about changes in shared repositories, allowing people to react when new information or new tasks are posted. The two main approaches are *push*: where collaborators receive an *explicit notification* when there is a change, or *pull* where there is no alert and participants must *actively visit* the repository to see whether a change has occurred. Both have disadvantages. Explicit change alerts in email may overwhelm people, leading them to ignore notifications. On the other hand, relying on participants actively visiting the repository may lead them to be unaware of new versions.

Despite successfully using shared repositories, our participants were still dependent on email alerts. Only one participant wanted to actively access the repository to detect updates. People were aware that pushed alerts contribute to email overload but accepted this as the price they must pay for not overlooking important content.

Such email alerts did not contain attachments, as this would undermine the entire point of using shared repositories. Emails often contained *repository links* however. Accessing the repository rather than an isolated email attachment has very different consequences. Accessing the repository meant that people can see *all* recent versions allowing them to straightforwardly find the most recent version. This contrasts with email versioning involving collating multiple versions across different emails, hoping that an email search has exhaustively found all relevant materials.

(p27) Rather than sending the document itself, which the challenge of that is that document is a snapshot in time, I'll send a link to a document. And now once you've sent a link to a document the people have the updated version of it. Then if there's an updated version they'll be able to see the new version and the old version; The real problem [with email] is that, you know, for all intents and purposes email is transitory. I mean, yes, I have a saved copy of it, but, well, which version of it?

In some cases alerting was supplied automatically by the underlying collaborative tool. Dropbox provides screen alerts when shared folders change. Again however, with one exception, users erred on the side of caution and also explicitly posted email alerts. Of course a major problem with increasing email traffic this way is that overloaded inboxes may lead participants to *overlook* the alert, and we discuss this problem later.

Task Management Involving Shared Repositories
However we saw multiple cases of where simple email alerts were insufficient. It was not enough to email collaborators that the contents of the shared repository had changed. Often more explicit actions and responsibilities were included to describe outstanding tasks. This participant points to the target document and describes what actions need to be performed on it.

(p14) you write an email saying 'here is a link to the Google document and here is what you are supposed to do: number one do this number two do that'.

Task management emails might contain links or a ContentMap detailing tasks and responsibilities, as well as how to access relevant content. Even when teams used dedicated workflow applications for task management, collaborators sometimes still provided an additional email nudge or clarification. Here a participant talks about how an email conversation clarifies the status of an outstanding task that had been delegated via a task management system.

(p9) if somebody is not following up with something, if somebody didn't notice something in [the task management system], I'll just send them an email linking back to the [workflow] ticket saying 'hey did you see this I think this needs attention' and then they can let me know that 'I saw that, it's not a priority because of whatever reason or thanks I've been meaning to get to that or..' sometimes people need a little extra push.

Social Finding
Finally and rarely email was used as a last resort if people couldn't find information in the shared repository. As we noted above, refinding problems were uncommon, but when they occurred, one tactic was to exploit transactive memory and email someone likely to know where the information was. However we found just two examples of this in our participant set, and both agreed that this was an unusual situation. One participant who was uninvolved in an early phase of a project wanted to locate relevant data and so emailed another collaborator who collected that data:

(p1) I wasn't heavily involved in the data collection and analysis, so I would find someone who was and say, "Where is this data?" and then get a pointer to it and access to it.

So to conclude, while email is being used less for distributing attachments, it retains multiple interdependent functions with shared repositories. All of these functions share a common weakness however; an overloaded inbox may mean that collaborators overlook critical emails, and email search does not guarantee that all messages relevant to the collaboration are reliably retrieved. We suggest technical implications that partially address this in the following section.

DISCUSSION AND CONCLUSIONS
Our findings address a paradox. Prior research fails to explain increased collaborative repository adoption, instead documenting teams' problems in co-organizing and retrieving shared information. In contrast, we present new empirical findings that identify practices that allow people

to circumvent these problems. We document novel practices that facilitate successful adoption of shared repositories, leading to a changed role for email. We explore the theoretical and technical implications of these new findings.

Strategies for Finessing Co-organization Problems

In contrast to prior work, we found that participants experienced few co-organizational problems. However our participants were not using keyword or desktop search to find information, even though this could potentially have removed the need to actively co-organize. How then were participants able to co-organize when prior studies show the opposite? We found they relied on strategies to co-organize including: *ContentMaps,* using other applications such as wikis, task management and Googledocs achieving co-organization by providing structured 'front ends' linking directly to shared materials. Other implicit strategies imposed predictable high-level structures based on expertise, task and tool affordances relying on shared knowledge between participants. Participants seemed to exploit transactive memory [28], being aware of coworkers' expertise, and delegating organizational tasks to experts rather than everyone contesting and being consulted on all aspects of organization. Furthermore both ContentMaps and implicit strategies seemed to be successful in promoting retrieval, with participants reporting few problems in accessing shared information.

How then can we explain these contrasts to prior work? There are at least two possibilities. The first is that prior studies looked at older tools *used in isolation*. In contrast in our work, participants addressed co-organization by *combining tools* in innovative ways, e.g. using wikis or task managers to structure information for their collaborators as ContentMaps. A second possibility is that we examined a different population carrying out very different tasks. Prior work looked at researchers often accessing long-term archival materials. In contrast, our participants were mainly focused on familiar short-term activities mediated by clear roles and responsibilities. It is interesting to note that the three exceptions in our sample who experienced finding problems were searching for long-term materials on projects that they had not been involved in directly. These observations and participant comments point to intriguing future questions about the scalability of current implicit strategies. Such strategies were used for relatively small active repositories, but it remains to be seen whether they can scale to larger, more complex settings.

The Changing Role of Email

Repository use was very closely tied to email practices and we also explored participants' use of email in collaboration. Consistent with prior work we found problems in using email for exchanging and locating multiple document versions. Greater use of repositories did not mean that email was abandoned, however. Instead it was used to *support* shared repositories. Email was used for alerting, task management and occasionally locating missing information. These uses of email are not problematic in terms of collaboration, as they provide social accountability, but they do exacerbate general email overload increasing the possibility that important collaboration-relevant emails get overlooked. We now explore ways of addressing this.

Design Implications

Our findings about synergies between tools and the changing role of email suggest several concrete ways that collaboration tools might be improved. We are aware that aspects of these solutions are available in enterprise contexts but not as standard features of common collaborative repositories like Dropbox or Drive. One approach would aim to improve access to, and management of, shared repositories, by making ContentMaps a standard repository feature. The power of ContentMaps was that participants saw task instructions and responsibilities linked directly to underlying content. ContentMaps are a useful way to reference shared content, so every repository could have a wiki-like structure describing repository content, along with direct links to that content.

We also saw the importance of implicit strategies using task or role information for organizing shared repositories. To support such strategies, tools like Dropbox, Drive, and shared folders might be modified to allow automatic tagging of files with roles or tasks. Tags could include derived metadata about content, including document creator, most recent editor, workflows and so forth. Although there are benefits to the simplicity of current tools, automatic tags could support enhanced views that extend simple folders (Dropbox) or flat lists (Drive) to mirror implicit strategies by showing tasks, owners and responsibilities.

Another important technical implication concerns improved integration of email and share repositories. One approach would be to modify email to support *active structured links* to repositories. We found that emails currently contain information about alerts, collaborative tasks, and pointers to shared content often using ContentMaps. Email clients might provide ContentMaps natively as a specific template. Templates could be automatically created containing links to shared content, making it easier for collaborators to share action plans about shared content. Such linking facilities are already provided in software development tools (IBM Rational Team Concert, MS Studio). Linking is also supported in a simplified fashion in Gmail, but regular email clients could be modified so that structured indexing is routine.

A different but complementary approach to email integration might be to populate shared repositories with communication information. Currently when you access Dropbox or Drive you don't see relevant email communications concerning that shared content. However such communications could be directly incorporated into the repository without having to leave it to access email or a

task manager. Again such facilities are provided in some software development contexts and research prototypes [22]. Such integration could be a useful addition to shared repositories such as Dropbox, Drive, or shared file systems.

Limitations and Future Work

Limits of the work are that the participant set is relatively small and homogeneous. It also needs to be supplemented with direct observation of teams to triangulate how strategies are enacted. There are also outstanding questions about the effects of shared repositories on work practices. Our work analyzed relatively small teams, but do successful techniques scale when collaborators are less familiar with each other's expertise and job responsibilities? Can ContentMaps succeed when collaborators have less shared knowledge? Other questions concern email. How will email practices change as email is increasingly used for coordinating around repositories rather than distributing attachments? When do people choose to share attachments rather than use shared repositories? How important is active leadership in the successful implementation and adoption of these tools?

In conclusion we have shown how and why current shared repositories are beginning to address the classic co-organization barrier. We have reported new uses of collaboration tools in the forms of ContentMaps and implicit co-organization that allow participants to work together effectively without needing to explicitly agree on collaboration structures. We have also reported how this has changed uses of email, and suggested ways that new technologies might improve existing tools.

REFERENCES

1. http://reviews.cnet.com/8301-13970_7-57571513-78/dropbox-clears-1-billion-file-uploads-per-day/
2. http://techcrunch.com/2012/06/28/google-drive-now-has-10-million-users-available-on-ios-and-chrome-os-offline-editing-in-docs/
3. Abrams, D., Baecker, R., & Chignell, M. Information archiving with bookmarks. *Proc CHI,* ACM Press (1998) p. 41-48.
4. Ackerman, M. S., & Halverson, C. Organizational memory as objects, processes, and trajectories. *Proc CSCW, 13*(2) (2004) p. 155-189.
5. Aula, A., Jhaveri, N., & Käki, M. Information search and re-access strategies of experienced web users. In *Proc WWW*, ACM Press (2005) p. 583-592.
6. Bannon, L. and S. Bødker Constructing Common Information Spaces. *Proc CSCW*, ACM Press (1997) p. 81–96.
7. Bellotti, V., Ducheneaut, N., Howard, M., & Smith, I. Taking email to task. In *Proc. CHI 2003,* ACM Press (2003), 345-352.
8. Bellotti, V., Ducheneaut, N., Howard, M., Smith, I., & Grinter, R. E. Quality versus quantity: E-mail-centric task management and its relation with overload. *HCI*, *20*(1) (2005) p 89-138.
9. Bergman, O., Beyth-Marom, R., Nachmias, R., Gradovitch, N., & Whittaker, S. Improved search engines and navigation preference in personal information management. *ACM TOIS*, 26(40) (2008).
10. Bergman, O., Whittaker, S., Sanderson, M., Nachmias, R., Ramamoorthy, A. The effect of folder structure on personal file navigation. *JASIST*, 61(12) (2010) p. 2426-2441.
11. Berlin, L.M., Jeffries, R., O'Day, V.L., Paepcke, A., & Wharton, C. Where did you put it? Issues in the design and use of a group memory. In *Proc. CHI 1993.* New York: ACM Press (1993) pp. 23–30.
12. Cutrell, E., Dumais, S. T., & Teevan, J. Searching to eliminate personal information management. *Communications of the ACM*, *49*(1) (2006) p. 58-64.
13. Dabbish, L. A., & Kraut, R. E. Email overload at work: an analysis of factors associated with email strain. *In Proc CSCW*, ACM Press. (2006) p. 431-440.
14. Ducheneaut, N., & Bellotti, V. E-mail as habitat. *Interactions*, *8*(5) (2001) p. 30-38.
15. Dumais, S., Cutrell, E., Cadiz, J. J., Jancke, G., Sarin, R., & Robbins, D. C. Stuff I've seen: a system for personal information retrieval and re-use. In *Proc SIGIR*, ACM Press (2003) p. 72-79.
16. Grudin, J. Groupware and social dynamics: eight challenges for developers. *CACM*, *37*(1) (1994) p. 92-105.
17. Jones, W. *Keeping Found Things Found.* Morgan Kaufmann, 2010.
18. Kushmerick, N., Lau, T., Dredze, M., & Khoussainov, R. (2006, July). Activity-centric email: A machine learning approach. In *Proc AAAI 21(2)* (2006) p. 1634.
19. Lutters, W.G., Ackerman, M.S., & Zhou, X. Group Information Management. In W. Jones & J. Teevan (Eds.), *Personal Information Management* Seattle: University of Washington Press. (pp. 236–248), 2007.
20. Mahmud, J., Matthews, T., Whittaker, S., Moran, T., & Lau, T. Topika: integrating collaborative sharing with email. In *Proc CHI,* ACM Press. (2011) p. 3161-3164.
21. Malone, T. How do people organize their desks? Implications for the design of office information systems. *ACM TOIS*, 1(1) (1983) p. 99-112.
22. Marshall, C., & Tang, J. C. That Syncing Feeling: Early user experiences with the cloud. In *Proc DIS*, ACM Press (2012) p. 544-553.
23. Nardi, B. A., Whittaker, S., Isaacs, E., Creech, M., Johnson, J., & Hainsworth, J. Integrating communication and information through ContactMap. *CACM*, *45*(4) (2002) p89-95.
24. Rader, E. Yours, Mine, and (Not) Ours. *Proc CHI*, ACM Press *(2009)* 2095-2098.
25. Russell, D., & Lawrence, S. Search everything. *Personal information management*, (2007) 153-166.
26. Shami, N. S., Muller, M., & Millen, D. Browse and discover. *Proc CSCW,* ACM Press (2011) p. 295-304.
27. Voida, A., Olson, J. S. & Olson. G. M. Turbulence in the Clouds. In *Proc CHI* ACM Press (2013).
28. Wegner, D. M. Transactive memory: A contemporary analysis of the group mind. In *Theories of group behavior* (pp. 185-208). Springer New York. (1987)
29. Whittaker, S., & Sidner, C. Email overload: exploring personal information management of email. In *Proc CHI*, ACM Press (1996) p. 276-283.
30. Whittaker, S. Supporting collaborative task management in e-mail. *Human Computer Interaction*, *20*(1-2) (2005) p. 49-88.
31. Whittaker, S. Personal information management: from information consumption to curation. *ARIST*, *45*(2011)p3-62.