# Managing work in team

*an overview*

A document from Open Lab

Updated for Teamwork version 3.1
Last modified on Tuesday, November 28, 2006

Teamwork is © Open Lab 2001 – 2006

Adobe®, the Adobe logo, Acrobat®, ReaderTM, and Macromedia Flash PlayerTM are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.
Sun, Sun Microsystems, Solaris, Java, JavaServer Web Development Kit, and JavaServer Pages are trademarks or registered trademarks of Sun Microsystems, Inc.
UNIX is a registered trademark in the United States and other countries, exclusively licensed through X/Open Company, Ltd.
Mozilla is © 1998- by Contributors to the Mozilla codebase under the Mozilla Public License and Netscape Public License.
Netscape® is a registered trademark of Netscape Communications Corporation in the United States and other countries. Netscape CommunicatorTM is a trademark of Netscape Communications Corporation that may be registered in other countries.
Microsoft®, Windows®, Windows NT®, SQL Server®, Microsoft Project®, and Internet Explorer® are registered trademarks of Microsoft Corporation.
Oracle® is a registered trademark of Oracle Corporation.
Informix® is a registerd trademark owned by IBM (International Business Machines Corporation)
Sybase® is a trademark of Sybase Inc.
FrontBase" is a trademark of FrontBase, Inc
Interbase® is a registered trademark of Borland® Copyright© 1994- Borland Software Corporation
MySQL is © 1995- MySQL AB under the free software/open source GNU General Public License (GPL).
PostgreSQL is copyright © 1996-2002 by The PostgreSQL Global Development Group
All other brand and product names appearing in this document may be the trademarks or service marks of their respective owners.

Find and try Teamwork at **www.twproject.com**

*To contact us:*

info@twproject.com
phone + 39 055 5522779
fax +39 055 582451
Florence / Italy

# Table of contents

# 1 Introduction

Welcome to Teamwork, the solution for daily project and workgroup management.

## 1.1 About this document

This document introduces Teamwork, and the main concepts dealt with in it. Having these concepts clear, it should be possible to use all of Teamwork's functionality.

Teamwork's detailed documentation is mainly in the videos, which collectively constitute a user guide. This document is not a replacement for the videos, which contain far more information. By "documentation" we mean the bundle of videos and supporting documents, which are this one and the installation guide.

Documentation is particularly helpful for project managers, which should be sure that all the mentioned concepts are clear; once this done, they could transmit the core ideas to users, say by having a meeting. Documentation can be used as a guide for educational meetings concerning software aided work management.

### 1.1.1 Thanks

We thank the various forum contributors and direct mail contacts who have helped in finding critical issues and features that were needed from version 2. We also thank Brendan Boyle for improving the English of this document.

## 1.2 What is Teamwork?

Teamwork is a solution for managing work and communication through projects in any field. Groupware and project management features are used in an integrated environment, from which you can coordinate and manage hundreds of open projects at once. Teamwork is easy to use, so that the entire team can contribute.

Teamwork is unique in that complex management structures get handled in a simple environment.

There are two main kinds of Teamwork users: those that set up the structure of the working environment (we'll call them *project managers*, as this is the custom), like setting the work areas and structuring the projects, and those that move in such environment, reading and inserting data of all kinds (we'll call them *users*). Moreover initially involved in Teamwork are also the system administrators, the technical staff.

Teamwork is

*(manage)* a software tool to *manage* work of groups of people,

and at the same time

*(communicate)* a tool for groups of people to get *coordinated communication* in order to work in a managed way.

This two-sided nature is its peculiarity, and what makes it an original application. Hence we didn't call it neither a *project management* application, nor a *workgroup* application. We call it a *work management* application.

A basic assumption is:

(*structure)* we can deal only with structured work

so one of our aims is

all work done gets structured, or *modeled*, in Teamwork

Teamwork is useful for companies that *already* have as aim the structuring of work; it just helps in achieving such goal.

Work gets structured in projects, and workgroups get implicitly defined by associating people to projects; in the following we'll articulate how to achieve this.

We find it strange to have project management software (e.g. that draws Gantt graphs) without a shared agenda; or work log without issue tracking. On the single feature, Teamwork can be beaten at times, but it gives an organic solution.

Teamwork contains several simple solutions, found over the years, which were not simple to find; some of them are an evolution from the obvious table-and-field work log management solutions, which were found to over simplify matters and at the same time generate too many fields, and were replaced with more mature models. I believe that there is where we may have left our competitors behind.
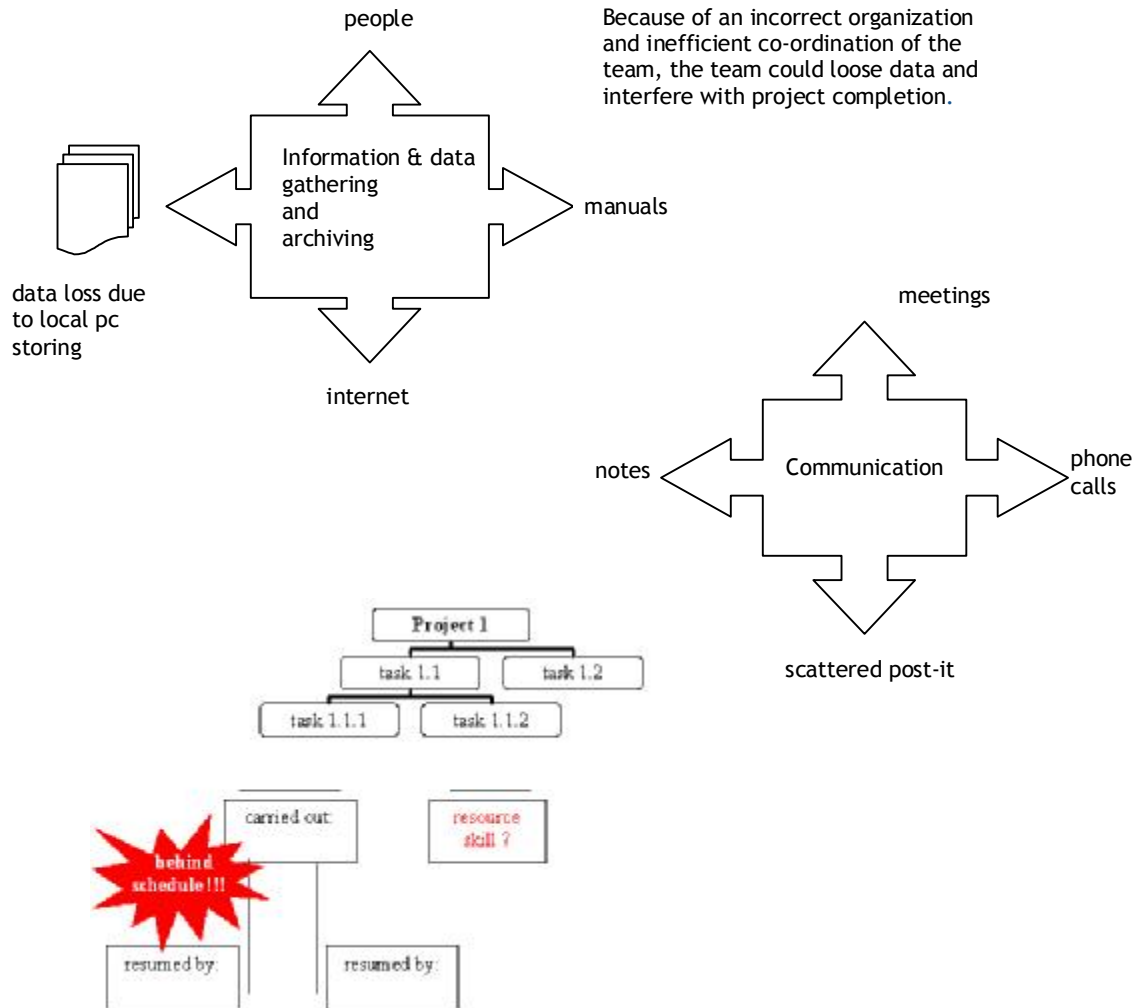
# 2 Aims and problems

## 2.1 Information flow

In working environments, data and digital information flow increases every day, as well as the need to solve different problems in the shortest time. Thus the increase of complexity emphasizes the limits of traditional project management tools, not able to face up to project and workgroup management at the same time, and with limited access features.

Teamwork is different because it has been deployed to structure a project and to support its development, splitting it into its different sub-components and tasks, with easy, dynamic and coordinated communication and data transmission in the workgroup. Traditional Gantt-centric applications do not solve the main problems.

## 2.2 Working in team

A project is carried out by a number of differently skilled resources, most of the time working in disparate locations, sometimes speaking different languages, but sharing a common goal: to achieve an agreed objective as efficiently as possible. Each resource should be well integrated into the team and efficiently co-operate with others.

Resources' inner quality and skills is obviously an important prerequisite, but no one must be essential to project final good outcome. Moreover, each team member should be involved, kept well informed and provided with the right tools to work out assigned tasks. It may happen that team members are required to interact efficiently, without having the tools to do so.

people

Information & data gathering and archiving

manuals

Because of an incorrect organization and inefficient co-ordination of the team, the team could loose data and interfere with project completion.

data loss due to local pc storing

internet

meetings

notes

Communication

phone calls

scattered post-it

Project 1

task 1.1

task 1.2

task 1.1.1

task 1.1.2

carried out

resource skill ?

behind schedule!!!

resumed by:

resumed by:

*Picture 1: some examples of dispersed information are shown*

By contrast, a correct workgroup organization should facilitate:

- real time transmission and sharing of data and information flow
- monitoring of different project sub-units, with automated reminders
- daily monitoring of tasks carried out by team members
- easy and fast communication among users

It is also clear that sharing of information must depend on a security policy, assuming that different skills and different responsibilities need different kinds of data access.

## 2.3 Requirements

Thanks to the adoption of a project and groupware tool, a company would expect to:

- be kept informed about each employee work log and job activities
- easily communicate with different project teams
- share information in compliance with the assumed standard
- rationalize and keep track of data and information flow
- rationalize, as final goal, its organization
- create assignments and set daily, weekly (or wider) priorities
- integrate existing workflow models, data sources and applications
- have a harmonic relationship between assignments and security settings

A worker would expect to:

- not to be hindered from doing usual work in usual time
- input and modify easily the work logs
- fast access to her/his assignments
- fast access to data concerning his task
- have a single *entry point* to organize his work in compliance with the human resource department
- preserve her/his personal computer habits and tools (email and chat client, browser, text editor etc.)
- have a "roaming" access from any pc, in compliance with the company standards
- customize the home page, the look and feel

As a consequence, the adoption of tools focused on teamwork should be consistent with single resource requirements and habits, in order not to interfere with her daily matters.

## 2.4 Approaching a solution

Many different factors contribute to reaching the goal of a project: thus, it is essential to use (availability) of a tool to coordinate, manage and monitor

- resources involved in its deployment

- tasks assigned to each resource
- data flow among authorized resources

Teamwork has been deployed to fulfill these requirements. It is a solution that can be adopted whenever you have to deal with an activity that can be split in different phases and among different users. That is, whenever you deal with a coordinate production.

In fact Teamwork combines features from project management, groupware and document management software, resulting in an innovative and peculiar application.

Thanks to the integration of document sharing and information flow tools, teams can access a constantly up-to-date body of information, in compliance with the matching grants.

Moreover, thanks to a personal and team shared calendar module, the scheduling of briefings and face-to-face meetings can be completely integrated with personal scheduling.
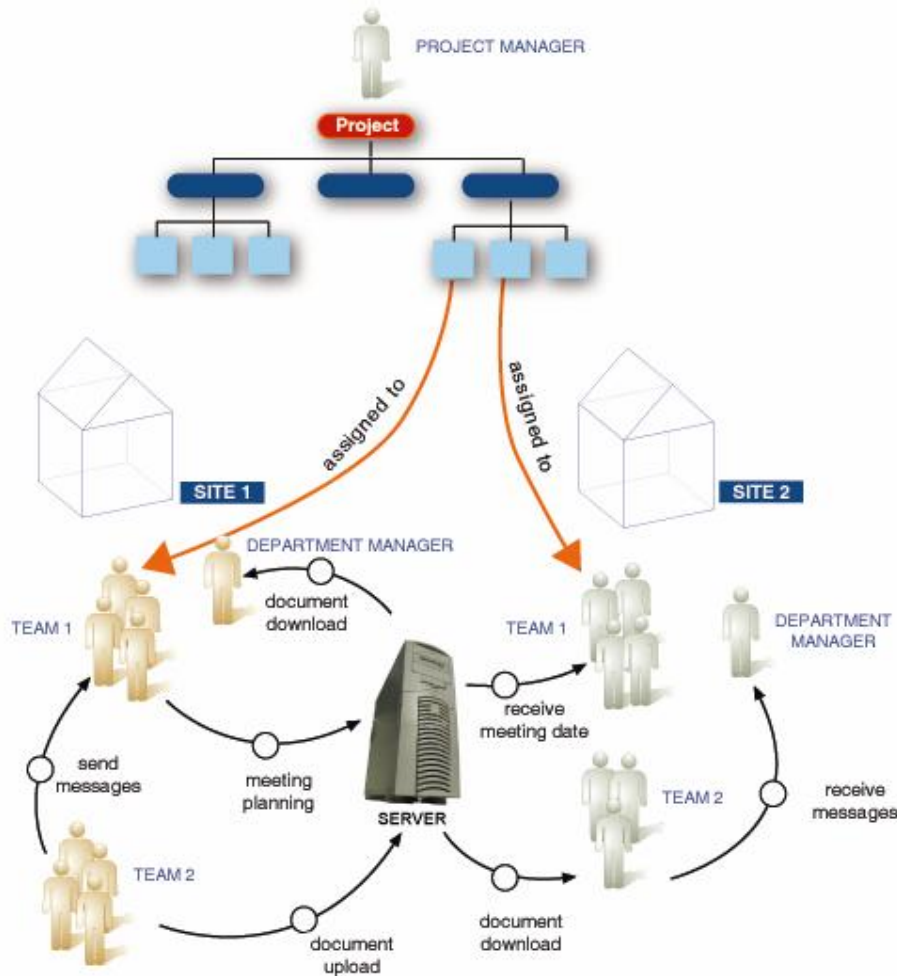
*Picture 2: managed work*

The adoption of web-based technology allows and enforces direct communication among team members, even if distributed in different district or countries. In fact Teamwork is totally internationalizable, also for non-Latin alphabets, and each user can set her language and date settings (also called "localization" settings).

Documents can be stored into a secure and centralized archive on dedicated servers. Actually, if the company already has a secure document organization, Teamwork can be configured so to totally integrate with such structure.

Each document will be accessible only to users provided with the specific permission for the specific document, with no burdens for administrators, as security is inherited from projects and resources. Automatic document versioning is supported.

Each user can easily send messages to the whole team or to a selection of members.

General reports, focused on the different objects and task in particular, help the project manager to constantly monitor the project state of progress.

Task assignments, work log, cost computation are just some of the features supported.

Integrating existing database data and servers is fast, being multi database and multi platform. Teamwork's open data structure and flexible workflow module make integration and customization easy, where for many systems it's expensive or impossible.

Open Lab is confident that you will find that the use of Teamwork greatly increases your productivity, improving at the same time the quality of work. You will be able to systematically gather a great volume of data, concerning tasks and issues; thus you will find how easy can be the management of information flow concerning a coordinate production cycle. Moreover everything you used to do with handwritten to-do lists (there is also a "to-do" active part, see the dashboard video) and single doc files will come together in a single application.

## 2.5   Simple and complex models

After the release of Teamwork 2 in 2003, several web-based application groupware applications have been published. Teamwork is in several aspects different from all the others, in particular for its basic philosophy.

A user evaluating work management software can easily confuse having to render a complex model, with feature bloating. The fact that Teamwork's interface is more complex then that of some of its supposed competitors is not due to feature bloat, but to the articulation of the concepts involved. Actually, our powerful interfaces are a sign of health.

Let's consider security problems for even a medium-small company, say with a hundred employees. We bet that no one can find a single example of such a company where everyone can see all the information floating in the company in the IT structure. But more: there will always be local exceptions to a simple role-based security model, because there will always turn up a particular project that has its own security.

Teamwork 2 had a role based security model, further refined by "areas", which were a way of completely separating say departments, in which the roles could be distributed. There could be users having different roles in different areas. But, as in all fundamentally role-based models, once you had a role, you would have it in the

entire area; so once you could read a task in say production, you cold read all tasks in production. So users complained again and again that they wanted a more fine grained security, as fine grained as a single task. And notice that this has to scale with respect to the numbers of projects, users and roles, for example when you do a search, so you have to express this by queries (HQL or equivalently, SQL) instead of navigating the objects in order to be performant, and you have to comply to the security model, which is non trivial. But for version 3, by leveraging the intuition of giving a role weight to assignments, we got all we had with version 2, plus fine grained security; and teams and structured work for free, too. All competitors we've seen so far don't even reach Teamwork 2's security model. And any (Team) work management application that doesn't solve the case of exceptions, simply cannot work.

So you must have a complex model, covering 100% of cases. What distinguishes a good model from a bad one is how it covers *limit cases*. Assuming that your model covers all above, and more, then we can talk about the interface ergonomics; but applications with a wrong model, will not fit the real world, however pretty their interface may be.

Also the reasoning of some competitors for separating applications is leaky. How could you have distinct applications for security and work management? And users are supposed to jump from one application to the other? Not very user friendly.

Interfaces for such applications cannot be simple; but they can be comfortable. And in Teamwork we have and are working hard on this (see the "Smart features" page, http://www.twproject.com/Smart_features.page). From release 3.0.0 to 3.1.0 the interface has been improved in hundreds of places, but the model in unchanged.

## 2.6 Want to know more

Many themes here presented can be treated more in depth. We list here a few sources of information.
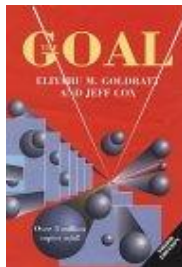
Literature on teamwork and groupware:

Jorg Geiger, Open-Source-Groupware Uberblick, Kategorisierung, Auswahl und Installation, Technische Universitaet Muenchen, 2003.

Matthias Leu, Contextual Collaboration and Integration for Community Support, Technische Universitaet Muenchen, 2004.

On work management:

The Goal: A Process of Ongoing Improvement, Eiyahu M. Goldratt, Jeff Cox, Gower Publishing Limited.

# 3 How Teamwork came to be

We'll consider the problems someone who wants to introduce "work management" in a team may meet. There is a typical path of try-and-fail which many follow, which we describe in the following.

Most of what we will say is valid in general for small teams. Typical work is:

- new development
- documentation
- maintenance
- support
- teaching/learning

We wish to get a bi-directional feedback between PM's and "workers", and want to see what's happening once the project gets going. In short, we aim at answering the following questions:

- what should they do today?
- what should I do today?

We won't answer the following question:

- which complex project scenario may fit imaginary data and CEO's wishes?

Traditional project management software is often meant for huge projects, hence has highly involved editors: but very few projects have the complexity of producing submarines, but have instead complex structural interaction needs.

## 3.1 First try: sparse applications

In order to try to answer our questions, introduce work management through software. In a first attempt, we use sparse applications (SPARSE):

- shared excel files, at least one for project
- isolated outlook/thunderbird instances
- use email for group events
- work log by memory
- chat and forum software
- paper post it

SPARSE' plusses:

- low adoption barrier (a relevant factor)

SPARSE' minuses:

- complex/impossible data aggregation
- work log data is never complete/verifiable
- complex group communication

## 3.2   Second try: use specific software

In trying to get better, we do a second attempt: use specific software (SPECIALIZED) the "usual" local project management tool, an open source bug track software, and client-server work log clients

SPECIALIZED' plusses:

- PM's have fun in drawing complex Gantt charts
- complete/complex issue tracking

SPECIALIZED' minuses:

- working team forced to pretend to be reading an unreadable Gantt chart
- complex/impossible data aggregation
- projects and bugs/features run in different circles
- users actually get to hate it

## 3.3   Alternatives

The situation can be summarized as follows:

| Existing solution | Alternative solution |
|---|---|
| Project on a file | Project on the web |
| Nice diagrams | Nice and real-time data |
| Groups defined by hand | Groups implicit in project |
| From PM one way to team | From team to PM and vice-versa |
| High adoption barrier | Low adoption barrier |
| Data in proprietary format | Data on most-loved DB |
| Hidden object model | Public sources of object model |
| Hard to integrate | Easy to integrate |

With Teamwork we aimed at producing an alternative solution.

## 3.4  Teamwork's solution

Introducing our solution (COMPREHENSIVE):

*a single open web based application that does project management, issue track, work log recording, group communication, agenda, related files classification, post it, forum/project history*

Teamwork has been developed as an attempt to satisfy this requirement.

COMPREHENSIVE' plusses:

- on our server, on our db, on my browser (on my Mac), in my language
- ergonomic interface for relatively complex cases
- data aggregation and validation is easy
- ease of integration with existing IT

To these core issues add a both friendly and powerful interface, with many features to make it easy to use.

COMPREHENSIVE' minuses:

- partial adoption due to possible conflict with existing legacy
- web based interface felt inferior to client-server

Recent releases can be seen as attempts to solve partially comprehensive minuses.