

Communication Paths in a Distributed Software Development Environment

Table of Contents

1	Introduction	3
2	Defining Communication – Why Bother?	4
3	Who’s Involved	5
3.1	Standard Roles	5
3.2	Groupings	7
4	Communication Responsibilities by Group	8
4.1	Sponsor	8
4.2	Product Manager/User Manager	8
4.3	System Architect	8
4.4	Program Manager	9
4.5	Project Manager	9
4.6	Development, QA, Technical Writing Managers	9
4.7	Development, QA, Technical Writing Leads	9
4.8	Development Engineers, QA Engineers, Technical Writers ..	10
4.9	Business Analyst	10
4.10	UI Designer	10
4.11	Subject Matter Experts	10
4.12	Technical Experts, Interfacing Systems/Organizations	11
5	Other Communication Paths	12
6	Examples	13
7	Conclusions	16

1 Introduction

Picture these situations:

Three engineers from different groups talking at lunch about a difficult bug that one of them had spent most of the night fixing. Nothing wrong with that, right? Just some colleagues talking informally about their work.

The Product Manager meets with the User Interface Design Group to decide about priorities for the next release. Once again, perfectly appropriate. Part of the job of the Product Manager is to establish release priorities. Meeting with each engineering group is one of the ways this gets done.

A Sales Account Manager asks an Engineer to make a change in the next release that a customer has asked for. Oops. We just crossed a line here. We would expect the Account Manager to advocate on behalf of a customer, but going straight to the Engineer is the wrong way to do it.

The Project Manager for the client calls an Engineer at the vendor firm to ask how a particular feature in the application works. The line just got crossed again. The Project Manager for the client should have a primary person, probably a Client Manager, who can field these calls for the client. It's not that the client can't get these questions answered; it's just that they should be answered in the most productive way possible.

These examples point clearly to the need for defining communication paths for project teams. But how can you define them in a way that encourages helpful informal communication and that also ensures necessary formal communication?

2 Defining Communication – Why Bother?

At one extreme you could view a project team as a network with potential connections from each member of the team to every other member of the team. But if everyone talked to everyone else, life would be chaotic and the team would never reach its goals.

At the other end of the spectrum you could have tightly controlled communication that defines exactly who can speak to whom. This approach allows for no flexibility in the system and is overly restrictive.

Finding the middle

The answer of course lies somewhere in the middle. What is that middle? Before answering the question, let's look at the difference between formal and informal communication. At Waverley we think of formal communication as any communication that causes a person to change work priorities, goals, or that affects productivity. Everything else is informal.

So according to this definition, formal communication includes obvious events like team meetings, client email, design reviews. It also includes any nonessential and lengthy calls to engineers by the client.

The way to find the middle is to **define communication paths**—not to the degree that they limit informal communication—but so that formal communication can proceed in a predictable and productive way.

The communication paths in distributed teams are controllable, even across remote locations, if everyone on the team understands and plays by the basic rules.

Early definition of communication paths will help avoid problems. As the project evolves, communication paths can evolve with them.

So how does a team go about defining communication paths? It starts by **defining roles**, then by looking at **groups of roles**, and finally at **paths between groups**.

3 Who's Involved

3.1 Standard Roles

Every team will have defined project roles. In distributed teams, these roles tend to be largely the same from project to project. In small projects, several roles are typically handled by a single person. Because industry-standard names and definitions have never been established. Those roles and brief definitions are provided below.

- **Sponsor:** The sponsor is the individual paying for the development project. This can be an I/T executive, a manager in the area that will use the project, etc. The sponsor is the final decision-maker when required but tends to be hands-off, relying on the Program Manager's status and communications to alert her to any issues.
- **Program Manager:** The main coordinator for the project. Status reports, risk analyses, planning, action items, regular meetings are all handled by the program manager. The Program Manager is the keeper of priority, status, and schedule.
- **Project Manager:** Each site will typically have at least one Project Manager. The Project Manager coordinates schedules, reporting, status, etc. for the groups at the site. (In smaller projects, the Program Manager often fills this role; if there is only one Project Manager no Program Manager is required.) The Project Manager manages deliverables from her team.
- **Product Manager/User Manager:** Depending on the situation, this role may be played by a number of different people in an organization, although only one lead would be assigned per project. This person is responsible for ensuring that requirements and functional specifications are generated and that the customer (represented by the Sponsor) is happy with the progress and results.
- **System Architect:** Responsible for the software architecture and design, and plans for how the software fits into a larger overall infrastructure. The system architect will also likely be involved in technology stack choices, approaches to system interfaces, data flow and database design.
- **Development Manager:** Manages the developers from a personnel perspective, is involved in issue resolution on the project as needed (may function as the Development Lead)

- **QA Manager:** Manages the quality engineers from a personnel perspective, is involved in issue resolution on the project as needed (may function as the QA Lead)
- **Technical Writing Manager:** Manages the technical writers from a personnel perspective, is involved in issue resolution on the project as needed (may function as Technical Writing lead)
- **Development Lead:** Manages the development team's workload and technical direction on a day-to-day basis.
- **QA Lead:** Manages the QA team's workload and technical direction on a day-to-day basis
- **Technical Writing Lead:** Manages the technical writing team's workload and technical direction on a day-to-day basis
- **Developer:** Designs and develops code for the project based on Team Lead's assignments
- **Quality Engineer:** Designs and executes test cases for the project based on QA Lead's assignments
- **Technical Writer:** Designs and creates documentation for the product based on Technical Writing Lead's assignments
- **Business Analyst:** Works closely with the SME and Product/User Manager to create requirements, use cases, and functional specifications for the project
- **UI Designers:** On large or UI-intensive projects, a separate UI design team may create wireframes or similar structure to specify the user interface.
- **Subject Matter Expert (SME):** Understands the product's end-user business. This role is sometimes filled by the product manager. SMEs are resources for the project team.
- **Technical Experts:** Typically adjunct to the project team, called in if and when their technical expertise is required to solve a problem or address an issue
- **Interfacing Systems/Organizations Representatives:** Representatives of areas—business, organizational, or application specific—that will interface with the end product. These representatives typically are called in when their expertise or approval is required, or when action is needed on their part. An example of an interfacing organization would be I/T.

3.2 Groupings

Although the number of roles defined above is large, from a communication perspective they resolve into a much smaller number of groups who operate similarly:

- Sponsor
- Product Manager/User Manager
- Program Manager
- Project Manager
- Development, QA, Technical Writing Manager
- System Architect, Development/QA/Technical Writing Lead
- Development Engineer, Quality Engineer, Technical Writer
- Business Analyst
- UI Designers
- Subject Matter Experts
- Technical experts, Interfacing systems/organizations

4 Communication Responsibilities by Group

The following communication responsibilities help provide a clear path. It becomes the responsibility of all groups to include the Program Manager, and then the responsibility of the Program Manager to synthesize and distribute all important information across the team.

4.1 Sponsor

The Sponsor communicates primarily with the Program Manager and the Product/User Manager. The Sponsor often sits in on team meetings and status calls, and may request updates or particular information.

The Sponsor should not communicate directly with anyone else on the team about the project without the involvement of the Program Manager and/or the Product/User Manager. The reason for this is not to hide information from the Sponsor; the problem is that most team members know what is happening in their own area, but may not know the overall picture and may not understand what activities and directives are operating at a higher level. Direct contact from the Sponsor is generally distracting (particularly to junior members of the team) and can result in unintended consequences such as a sudden change of direction by the team member based on casual comments from the Sponsor.

4.2 Product Manager/User Manager

The Product Manager works directly with her local team, particularly the Project Manager and the Business Analyst, to make assignments and monitor progress. The Product Manager also communicates directly with the Sponsor (depending on the organization this responsibility may fall to the Program Manager). Most information requests for the larger team from the Product Manager are routed through the Program Manager; at a minimum, any communication with members of the larger team copy the Program Manager.

4.3 System Architect

The System Architect interfaces with the sponsor, the program manager, the development leads and the product management team. The System Architect is typically responsible for an overall design and implementation strategy and considers issues such as scalability, operational efficiency, data modeling and database requirements, technology stack and

frameworks necessary, etc. The System Architect will help guide the team and resolve important questions as they arise. All interactions with the System Architect should include or notify the Program Manager.

4.4 Program Manager

The Program Manager is the primary point of contact for the team, delivering regular status updates, risk assessments, etc. The Program Manager consolidates and directs communication within the team. The Program Manager should be included in all inter-group communications, both within a single site and across sites. The Program Manager tracks action items and escalates issues for resolution, and ensures that communications involve all needed parties.

4.5 Project Manager

Each Project Manager is the primary point of contact for a specific group (often for a geographic group) on the project. The Project Manager collects information from her own group and forwards that information to the Program Manager (at a minimum). The Project Manager tracks status and issues within her own group, and forwards cross-group issues and needs to the Program Manager for tracking and resolution.

4.6 Development, QA, Technical Writing Managers

The Managers make personnel decisions regarding who is on the team and what workload they can manage. The Managers change personnel on the team according to project needs—skillets, headcount, performance, etc., notifying the appropriate Project Manager, who notifies the Program Manager. Any personnel issues are escalated to the Managers by the Project Managers; the Program Manager reports inter-group issues to the appropriate Project Manager and tracks the resolution.

4.7 Development, QA, Technical Writing Leads

The Leads manage communication amongst their own teams, reporting any personnel issues to the Manager and often the Project Manager. The Leads work closely with their respective Project Managers and other group leads, and often work with leads in other areas. Any communication across project groups includes the Project Managers of both groups as well as the Program Manager. Leads communicate status to the Project Managers for incorporation

into the group's status and schedules. All requests for status as well as work products must flow through the appropriate Program Manager and Project Managers. Leads handle issues within their own teams and work with their Project Manager to handle issues within the group; cross-group issues are escalated to the Program Manager through the Project Manager.

4.8 Development Engineers, QA Engineers, Technical Writers

Line engineers and writers communicate primarily within their own team. Communication with other teams within their group typically involves the leads for topics of larger importance (not casual questions or confirmations).

4.9 Business Analyst

The Business Analyst is primarily a representative of the Product/User Manager. Under direction from the Product/User Manager, the Business Analyst communicates directly with Subject Matter Experts, and works with the local Project Manager as well as with the Program Manager (all communications with the Program Manager are typically noted to both the Product/User Manager and the local Project Manager).

4.10 UI Designer

The UI Designer communicates mostly with the Program Manager, Product/User Manager, and Business Analyst. She often works with the SME, but with the participation of one or more of the three main contacts. Questions from Development regarding UI can be handled directly, but the Program Manager and associated Project Manager(s) are also involved in the interaction.

4.11 Subject Matter Experts

Subject Matter Experts communicate primarily with the Business Analyst. Other members of the team may need information from the SMEs, but typically the only ones other than the Business Analyst that communicate directly with the SMEs are the UI Designers. All communications with SMEs from anyone other than the Business Analyst within the group typically copy the Business Analyst. If the communication is from outside the group then the associated Project Manager and the Program Manager should also be copied.

4.12 Technical Experts, Interfacing Systems/Organizations

Technical experts and interface representatives are usually contacted by a technical lead, project manager, or business analyst. Although they may be in meetings with more extended project groups, their primary communication path is the person who established initial contact. That team member is responsible for communicating information from the expert or interface to the appropriate members of the project team, copying the Program Manager, Project Managers, and Product/User Manager at a minimum.

5 Other Communication Paths

One-on-one communication is not the only channel for receiving and disseminating information in a project team.

Project teams have regular project status meetings. In small teams, all team members usually attend; in larger teams leads, managers, project managers, the program manager, the product manager/user manager and business analysts attend. Bug triage, requirements and UI review, and similar regularly scheduled meetings include all affected team members.

There are also the usual intra-team meetings, including department meetings, code and requirements reviews, etc.

All of these standard meetings should be defined and scheduled on a regular basis in order to communicate effectively across the team. For geographically dispersed teams, regular face-to-face meetings should be scheduled as well to maintain a good working relationship.

6 Examples

6.1 Small Project Team

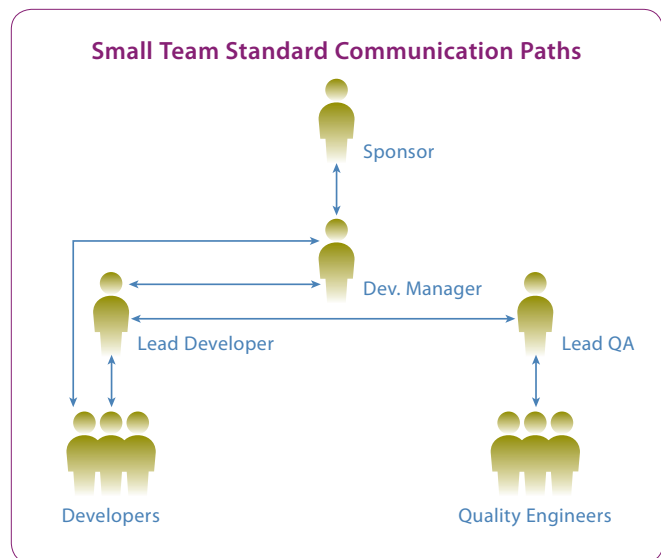
Small projects usually consist of teams from one to five developers. Typically, the development manager can handle all interfaces to other systems, if necessary, and acts as the product manager and program manager. The sponsor is critically important to the successful implementation of the project.

6.1.1 Usual Team

- Sponsor
- Lead Developer
- Developer(s)
- Lead QA
- Development Manager

6.1.2 Optional Team

- Product Manager
- Project Manager
- Additional QA
- QA Manager



6.2 Medium Project Team

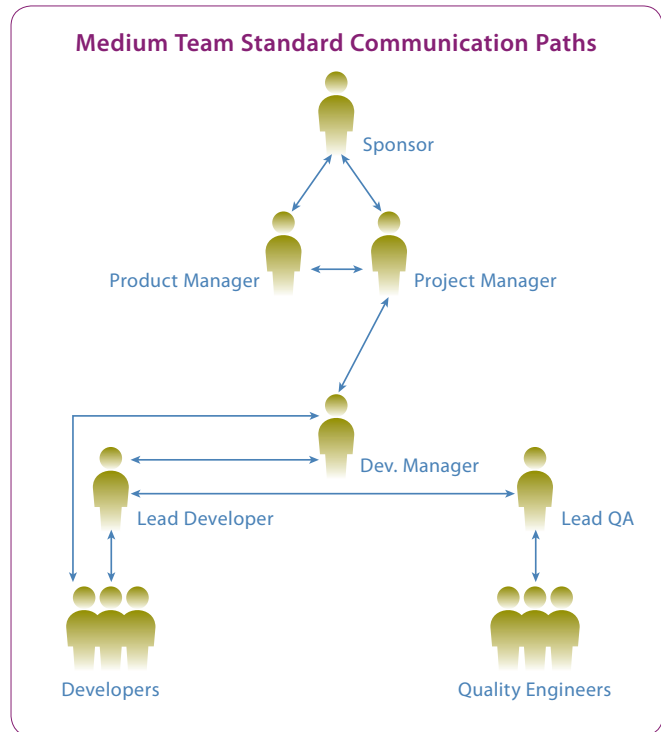
Medium projects run from a handful of developers to perhaps a team of 20 or so developers. They usually are larger in scope and cost, but do interface as heavily with other functions in a larger organization.

6.2.1 Usual Team

- Sponsor
- Lead Developer
- Developer(s)
- Lead QA
- QA engineer(s)
- Product Manager
- Project Manager
- Development Manager
- QA Manager

6.2.2 Optional Team

- Program Manager
- System Architect
- UI Designer
- SME(s)
- Technical Writer(s)
- Other technical experts/
Interfacing Systems Experts



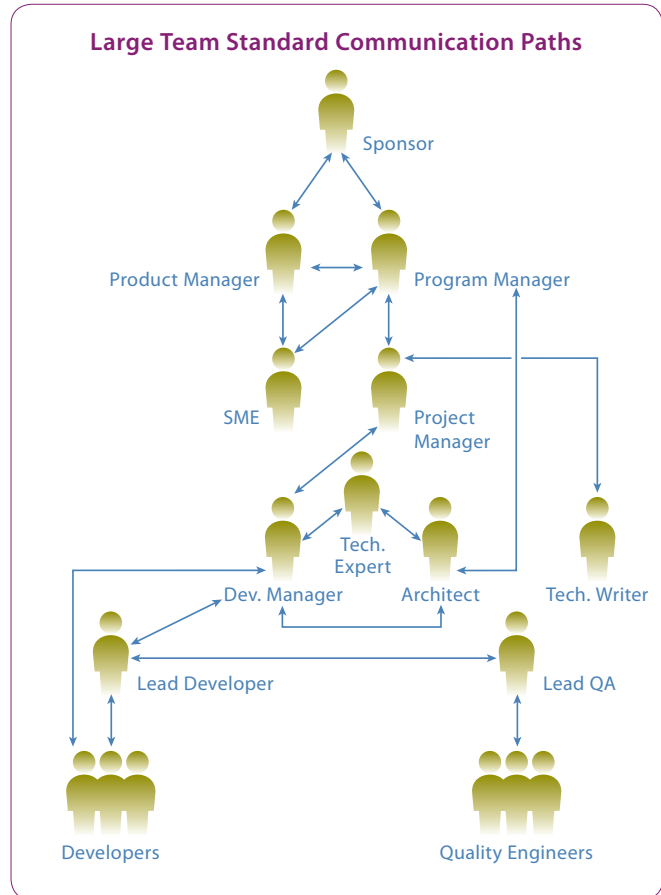
6.3 Large Project Team

6.3.1 Usual Team

Sponsor
System Architect
Lead Developer
Developer(s)
Lead QA
QA engineer(s)
Product Manager
Project Manager
Development Manager
QA Manager
Program Manager
SME(s)
Technical writers
Other technical experts/
Interfacing Systems Experts

6.3.2 Optional Team

UI Designer
Business Analyst



7 Conclusions

In distributed teams the definition of standard communication paths is critical. This helps direct the flow of information, ensuring that all team members have the same data concerning status, direction, and issues.

In these communication paths there are several central directors, primarily the Program and Project Managers, who ensure that the correct team members are involved in or apprised of any critical communications.

Other than limiting Sponsor or other executive access to individual team members (such access tends to be extremely disruptive and difficult to manage) the communication paths defined are not designed to limit information exchange. Instead, the paths are defined to direct and control information exchange to ensure that every team member is fully informed and that all team members are operating under the same parameters toward the same goals.