

Using Productivity Metrics to Manage Documentation Projects

Glenn M. D'Amore
Automatic Data Processing, Inc.
glenn_d@nac.net

Overview

Today, with constant pressure to justify every resource dollar, corporate management demands proof that a technical writing staff is a smart investment. One way to prove this is to show how your documentation provides value-added services to your organization.

The benefits of your documentation are often measured by factors such as its clarity, how well it meets client needs and expectations, if it improves client satisfaction or retention, and if clients would recommend the product because of the documentation. (By client, I'm referring to the audience for your documentation, whether inside or outside your organization.)

Surveys, feedback forms, focus groups, and usability tests help you determine the value-added benefits of your completed documentation. Nevertheless, they don't help justify the *development costs* or tell you if your team is *efficient* and *productive*. Effectively managing these three items gives you another way to prove your value.

Development costs include salaries, fees, training, hardware, software, and overhead. Every project has an acceptable threshold or budget for these costs. Regardless of the budget, project managers will try to keep these costs to a minimum. You can determine your project costs objectively if you track the associated purchases and the time you spend on tasks. Purchases can be tracked easily, but time reporting requires a system (manual or electronic) and your team's willingness to record their time accurately.

Efficiency is a factor of both your development costs and your productivity. Produce more or decrease costs and your efficiency rises. Decrease productivity or increase costs and your project efficiency falls.

Permission to make digital/hard copies of all or part of this material for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication and its date appear, and notice is given that copyright is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires specific permission and/or fee.

SIGDOC 97 Snowbird Utah USA
Copyright 1997 ACM 0-89791-861-4/97/10...\$3.50

To determine productivity you need to measure your output. This is especially true when you deliver information across varying media such as paper, online help, or the Internet. There are also challenges when your documentation is highly graphical and when your help or web pages contain numerous jump links. Methods such as pages, panels, or topics per week are often used to describe productivity. At ADP Information Development Services, we've developed a custom measurement called a "publication unit" that we use to track productivity.

Once you understand the development costs, efficiency, and productivity of your projects, you can begin to answer the following questions:

- How much cost is attributed to a specific project?
- How productive is the team?
- Can I compare productivity across projects?
- Are current development processes efficient?
- Can I use this data to repeat successful projects?

Answers to these questions prepare you to sell your services because you are managing all aspects of your project, not just your deliverables. In other words, you are treating information development as a business and not just another corporate service.

Costs and Their Significance

Put on your technical writer's hat.

You work hard and produce quality work. Like most writers in the profession, you sweat the slightest details over misplaced modifiers and split infinitives. Hopefully you're on speaking terms with your editor. And what about the never-ending functional changes? You know, those minor, undocumented changes that marketing and development always forget to tell you about. Do you lose sleep over these? When was the last time you received completed software or specifications when you actually needed them?

Sound familiar? If you're like most writers I work with, you probably roll up your sleeves and deal with these issues. That's what makes you a strong, professional writer. Writing is your job, your career, and what you love doing. Isn't it therefore your goal as a technical writer to deliver the highest quality documentation possible? The answer is yes, but it's not your only goal.

Now put on your manager's hat, and let's talk business, starting with the following statement:

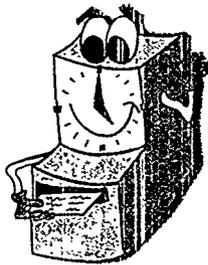
An effective technical writing project meets your clients' needs at an acceptable cost.

To be effective, you can't just deliver an outstanding information product. You must also deliver it within the budgetary constraints of your project. This is where productivity metrics can shed light on the business side of your team's technical writing processes. (To help you understand what I mean by team, a typical ADP documentation project team consists of several writers, an editor, a publisher/graphics designer, and a project leader.)

Time Reporting

Productivity metrics is founded on two principals: time and output. At ADP Information Development Services, we currently use a Windows-based software product called Innate TimeSheets (by Innate Management Systems, Limited) to record project and non-project time electronically.

In practice, each associate spends about 5 minutes each day recording how many hours (rounded to the ¼ hour) he or she spends on project and non-project tasks. Innate TimeSheets uses Microsoft Access as its database, so it's very easy to report and query on the data.



When we started piloting electronic time reporting two years ago, we tracked our project time against 40 task categories. Through trial and error, and some sanity checks, we've narrowed this down to eight categories for every project:

- Management/Administration (including meetings)
- Research and analysis (including planning)
- Development (text and graphics authoring)
- Editorial and technical reviews
- Production (such as publishing and help compiles)
- Infrastructure support (hardware/software setup)
- Training (for the product and development tools)
- Rework (revisions due to requirements changes)

With these eight categories, we have a better understanding of how much time we spend on each of our project tasks. The data is extremely useful when we estimate project needs and staffing requirements, such as how much training time to consider and whether to include a full-time editor on the team. It also gives us quantifiable data to back our estimates and staffing projections. Simply maintaining the numbers is usually

sufficient to fend off any inquiries, especially when most groups still use "best guesses" or "SWAGS" for their estimates.

In addition to these task categories, there are non-project categories as well, such as the following:

- Vacation time
- Personal time
- Sick time
- Jury duty
- Staff meetings

These categories factor into project availability, which I'll discuss later in this paper.

Electronic time reporting works like this. Associates complete an electronic time sheet at the end of the day or at various times throughout the day. Either on Friday afternoon or Monday morning, associates submit their time sheets electronically to their supervisors. The supervisors then approve and forward the time sheets to a system administrator.

Once a month, all hours are consolidated into a Microsoft Access database. Using custom queries and reports that we've developed and revise as needed, we can quickly analyze the data we want to track—such as hours per project, task, and person. We also consolidate our report data into a PowerPoint slide show presentation that we submit as part of our department's monthly report.

The rollout of electronic time reporting was a little rocky at first. While the software is easy to use, the idea was a little unsettling for associates who were not used to filling out time sheets. We had to convince them that we were not trying to see how hard they were working—we just wanted to learn more about where they were spending their time.

After a month or two, the initial apprehension to using the system disappeared. From that point on, our only issues have been an occasional late time sheet or a misunderstanding of the categories. For the most part, associates have fully accepted and are comfortable with the concept of time reporting.

Monthly Report Samples

To give you an idea of how we present our data in a monthly report, the following page contains sample Microsoft PowerPoint and Excel illustrations from actual reports.

Below is how we summarize the productivity of a completed project. I'll discuss more about publication units and resource costs later in this paper.

HR/Perspective Version 2.0 Rollout

- Deliverables
 - 1.3 Mb online help file
 - 3 guides
 - Acrobat PDF files
- 877 publication units
- 2,735 resource hours
- \$98,460 resource cost
- 12.8 units per week
- \$112 per unit

Figure 1: Productivity summary

This is how the 2,735 hours were allocated by task.

HR/Perspective 2.0 Hours by Task

- Research & Analysis 736
- Development 540
- Production 474
- Editing 447
- Management & Admin. 317
- Training 91
- Infrastructure Support 70
- Rework 60

Total Hours 2,735

Figure 2: Hours by task summary

This chart shows a 3-month snapshot of how our output was split between audiences (external clients and ADP regional support centers) and delivery mediums (online and paper).

IDS - PC Products	FY97			Plan			
	FY96 Act	FY97 Plan	Act YTD	Apr-97	May-97	Jun-97	Next Mth
Releases	72	50	59	7	3	6	7
Finished Publication Units	3953	6660	5743	617	488	78	150
Client Deliverables	63%	70%	74%	4%	100%		80%
Region Deliverables	37%	30%	26%	96%		100%	20%
Online Deliverables	67%	47%	41%	96%	25%	100%	29%
Paper Deliverables	33%	53%	59%	4%	75%		80%

Figure 3: Output distribution

This chart shows a 3-month snapshot of staffing levels within the department.

IDS - PC Products	FY97			Plan			
	FY96 Act	FY97 Plan	Act YTD	Apr-97	May-97	Jun-97	Next Mth
Current FTE Staff				22.0	22.0	22.0	29.0
Open Requisitions							3.0
Hired but not Started							
Approved FTEs		22.0		22.0	22.0	22.0	23.0
LOA and Disability				1.0			
Actual FTE Staff on Hand				21.0	21.0	22.0	29.0
Coops		2.0					1.0
Consultants				1.0	1.0	2.0	2.0
Actual Staff on Hand				22.0	22.0	24.0	23.0

Figure 4: Staffing levels

This chart shows a 3-month snapshot of items such as productivity factor, costs, rework, and project availability. I'll discuss more about these items later in this paper.

IDS - PC Products	FY97			Plan			
	FY96 Act	FY97 Plan	Act YTD	Apr-97	May-97	Jun-97	Next Mth
Productivity Factor	11.2	12.0	8.3	8.8	4.7	13.6	12.0
Total FTE Project Costs	\$2,931,669		\$353,874	\$101,088	\$126,288	\$83,388	
Production Unit Cost	\$127	\$129	\$173	\$164	\$310	\$188	\$120
Sick Time	1310		933	49	83	44	
Overtime	2894		2331	285	297	281	
Unplanned Rework Hours	1153		922	150	36	51	
Project Efficiency	85%	85%	87%	91%	82%	85%	85%
Project Availability	78%	80%	77%	82%	74%	81%	78%

Figure 5: Productivity, efficiency, and availability

This chart shows a 3-month composite of all completed projects by percentage of work devoted to each project task.

IDS - PC Products	FY97			Plan			
	FY96 Act	FY97 Plan	Act YTD	Apr-97	May-97	Jun-97	Next Mth
Completed Proj Allocations	100%	100%	100%	100%	100%	100%	100%
Research & Analysis	20%	20%	18%	12%	19%	42%	20%
Development	33%	30%	25%	33%	23%	23%	30%
Editing and Tech Review	6%	6%	13%	9%	9%	2%	5%
Production	15%	15%	15%	15%	23%	9%	15%
Mgt and Admin	15%	15%	15%	17%	11%	4%	15%
Training	4%	5%	4%	2%	2%		5%
Infrastructure Support	3%	5%	3%	3%	4%	1%	5%
Rework	5%	5%	4%	3%	9%	19%	5%

Figure 6: Task allocations

Some Insights of Time Reporting

We found it enlightening to uncover how much time is really available during a given workweek for project activities. We use this data when scheduling and planning staffing estimates for new projects. By tracking all project and non-project hours, we can determine a department average for how much time is available to work on project tasks. Simply stated, project availability is your standard corporate workweek (say 40 hours) minus non-project activities and unavailable time. These are defined as follows:

Non-project activities = Time at work that isn't devoted to a specific project, such as administrative activities and staff meetings

Unavailable time = Time away from work such as vacations, sick time, LOA, and jury duty

At ADP Information Development Services, our current project availability rate is 77 percent. This means that when we estimate projects, we add 23 percent to our time estimates to account for the non-project activities and unavailable time. Another way to state this is that there are only 31 hours (77 percent of 40 hours) per person in the standard workweek available for project tasks. It makes quite a difference when you include this factor in your schedules, because it gives you a more realistic estimate.

Incidentally, in project management circles, the terms Global Productivity Factor (GPF) and Non-Project Loading Factor (NPLF) are also used to describe non-project and unavailable time. These terms involve formulas but the end results are the same as I've described—they tell you how much time is available for project work.

How Much Does a Technical Writer Cost?

At ADP Information Development Services, we currently use \$36 per hour as the fully loaded hourly rate of full-time associates. Fully loaded means that we've accounted for salary, benefits, equipment, facilities, and so forth—the cost to the company, not the hourly rate of the writer.

If you're working with consultants, especially offsite consultants, you can use their true hourly rates instead. If you're really diligent, you can also compute your actual hourly costs based on the salaries of your team and a typical loading factor of 25 to 50 percent (the higher the percentage, the higher the costs and the more conservative your estimate).

Once you determine an hourly cost, you can multiply total project hours expended by cost per hour to determine total project staffing costs (for example, 2.735 hours x \$36/hour = \$98,460).

We also find that tracking hours spent on rework tasks is very useful. We define rework as any documentation revisions caused by unplanned functional changes that "creep" into the software. Rework is costly because it causes changes late in the product development cycle. The later the changes, the more costly the impact. While most projects incur some rework, efficient, well-run projects keep it to a minimum. (For more information on the high cost of fixing problems late in a product's life cycle, see "Adding Value as a Professional Technical Communicator" by Janice (Ginny) Redish in the February 1995 edition of the *Journal for the Society for Technical Communication*.)



Because ADP follows a project life-cycle development process, in theory, no functional development is unplanned. In reality, life isn't this simple. On some projects rework causes us hundreds of hours of repeated effort. This past year we devoted 922 hours (close to 6 months) of resource effort addressing rework. We also devoted 2,931 hours to overtime, partly due to rework and partly due to excessive workload.

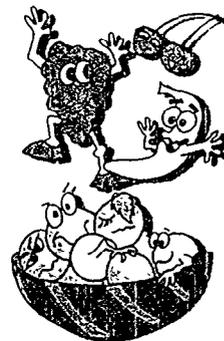
Time reporting allowed us to quantify our rework and overtime hours and hire additional staffing while other areas were going through a period of "right sizing." It also gave us ammunition to push back on continual marketing and product development changes because it made the impact of these changes more visible throughout the organization.

What is Your Productivity?

Time reporting alone will tell you how much time you're spending on project and non-project tasks, but without quantifiable and objective output, you won't know your true productivity.

At ADP Information Development Services, we wanted to come up with a standard metric for measuring our productivity, but found it difficult to compare output between our paper and online deliverables. We also found it difficult to account for graphics and illustrations in our productivity numbers because we wanted to include a factor for their varying complexity.

Our solution was to develop a unit of measurement that we call a publication unit. It would take a separate paper to go into the details of this unit, and our formula is still being refined to iron out some kinks, but let me at least describe its purpose. The publication unit serves as a generic unit of measurement for all documentation we produce, regardless of its medium, complexity of the information, or complexity of the design



Online and paper pages, panels, and topics are all weighted according to our publication unit formula and converted into standard units. In our department you don't hear phrases like pages per week; we talk in terms of "pub units" per week. It's as if we've taken a basket of mixed fruit and turned it into a basket of oranges.

At the completion of every project, the lead writer or writers use the publication unit formula to derive their final output. Since we've recorded all time against the project, we can easily determine the per unit cost and overall costs for the project. At this point we also factor in specific hardware and software purchases, training costs, and so forth, to capture total project costs.

We still keep count, however, of how much online versus paper documentation we produce, because we always try to minimize paper if possible. We've found through our metrics that we can develop online documentation in 60 percent of the time that we need to develop an equivalent amount of paper. So, if our clients want online information in our products as opposed to paper guides, we're happy to provide it.

Although we use publication units at ADP, industry measurements such as pages and panels per week would likely suffice for the purposes of most writing groups. For more information on decision factors for your estimates, refer to the information on the Dependency Calculator in the book *Managing Your Documentation Projects* by JoAnn T. Hackos.

Back to the Original Questions

Let's go back to the questions stated on the first page of this paper and see how we can answer them.

How much cost is attributed to a specific project?

The total project hours you record and the standard hourly cost you use lets you calculate resource costs. Then, by including purchases associated with the project, you can derive your total costs. (For example, 2,540 hours x \$36/hr + \$12,000 in hardware and software + \$6,000 in training = \$109,440.)

Remember that you reuse much of the hardware and software you purchase. The skills learned during training also benefit future projects. For these reasons, you should apportion only those costs that are actually attributed to the project. Although subjective, take your best shot at an estimate.

One rule of thumb is to assume that your purchases and training have a 2- to 3-year useful or relevant life (probably a valid assumption given the pace of technological change). Rather than simply apportioning the expenses equally over 2 to 3 years, give more weight to the initial project by realizing most of the expenses up front. Accelerated depreciation formulas, such as sum-of-the-years' digits, provide a useful analogy for doing this. Using this method, if you assume a 3-year life, apportion 50 percent of the costs to the initial project. If you assume a 2-year life, apportion 67 percent of the costs to the initial project. (Refer to any basic college accounting text for more information on depreciation methods.)

How productive is the team?

If you divide your total project hours by your workweek (say 40), you come up with total weeks for your project. You can then divide your total output (such as publication units, pages, or panels) by the total weeks in your project to derive a units per week figure. (For example, 230 units / 16.5 weeks = 13.9 units per week.)

Can I compare productivity across projects?

Once you start collecting and recording productivity by project, you can start comparing your projects. Again, if you can work with a unit of measurement that has already factored in issues such as complexity and delivery medium, you should be able to compare your productivity objectively across projects.

Are the current development processes efficient?

Look at your data. If some projects lag behind in productivity, find out why. The reason may be as simple as the experience level of the writers or lack of adequate training. There may also be process-related issues that you can correct. If you need to justify training dollars, presenting your management with tangible data gives you a better chance of getting what you need.

Can I use this data to repeat successful projects?

Through a better understanding of your productivity and efficiency, you can start to use successful projects as role models. This increases the likelihood that you'll repeat successful projects over and over again.

Summary

This paper presents some insight into the business of information development. As a business, we must strike a balance between meeting our client's needs and determining acceptable costs for the project.

Information deliverables are a component of a client product that must be managed from a revenue and expense perspective. Our revenues are the positive results that our organizations receive due to our documentation. Our expenses are the costs of our development. Both are important, and both must be managed if you are to be truly effective in your business of information development.

Trademarks

ADP is a registered trademark of Automatic Data Processing, Incorporated.

Innate TimeSheets is a trademark of Innate Management Systems, Limited.

Microsoft, Microsoft Access, and Microsoft PowerPoint are registered trademarks of Microsoft Corporation.

Microsoft Excel and Microsoft Windows are trademarks of Microsoft Corporation.

References

1. Hackos, JoAnn T. (1994) *Managing Your Documentation Projects*, John Wiley & Sons, Inc.
2. Redish, Janice (1995) *Journal for the Society for Technical Communication*, Volume 42, Number 1, February 1995.
3. Cartoon illustrations by Karin Hansen, Automatic Data Processing, Inc.