

# Outside Programming Services: Body Shops versus Contract Development Firms

Conrad Weisert  
June 1, 1998

More and more organizations are engaging outside contractors for major application development or enhancement projects. Sometimes such engagements work harmoniously and end happily; sometimes not. When they fail, the cause is often rooted in misunderstandings of roles and unrealistic expectations.

Such outside contractors fall into two categories:

- professional *software development firms* and
- *body shops*.

Each has its role, and a clear understanding of the differences between the two is crucial to a successful outcome.

## **Not consultants**

Clients sometimes call all vendor people *consultants*, but that's rarely what they are. A real consultant *consults* on aspects of an organization's performance. He or she may diagnose problems, recommend solutions, participate in implementing those solutions, provide ongoing advice ("mentoring"), or even write a little software, but the consultant's primary role in the organization is not to develop software.

True consultants can play valuable project roles when an organization undertakes its first project that will exploit some new methodology or technology, such as client-server architecture or object-oriented programming. An experienced expert can not only give essential direction and mentoring to the rest of a project team, but should also be a major contributor to the client's eventual body of permanent standards and guidelines.

But this article is about two different kinds of *developers* (or *contract programmers*), not consultants.

## **Body shops and brokers**

The term *body shop* carries a rather unprofessional image that's often but not always justified. Nevertheless, many situations call for just the services that a body shop (or "temporary agency") offers, and a shrewd manager can exploit those services effectively. For example:

- Most application system *maintenance* is well suited to a body-shop contract. The recent demand for coders to work on the so-called "year-

2000 bug" is being met largely and usually effectively through body shops.

- Many projects being managed and largely staffed internally need some specialized technical skill for a short time. Such needs may be met by temporary staff from a body shop.

A special kind of body shop is the *broker*<sup>1</sup>. Brokers keep few, if any, professionals on their permanent staff, preferring to engage individual *subcontractors*<sup>2</sup> when they land a contract and then shed them when the contract ends.

A common mistake is hiring a body shop or a broker when you think you're engaging a professional software development firm. Since few companies voluntarily admit to being body shops, you'll have to do some digging to determine which kind of firm you're dealing with.

Note that the difference isn't one of size. Both kinds of company range from a handful of people to staffs of hundreds.

This article describes some indicators that distinguish between the two. While few organizations are 100% one thing or the other, you can get a good idea which category an organization leans toward by looking at those indicators.

## **Infrastructure indicators**

A professional software development firm relies on a solid base of well-developed support functions. Those functions are rudimentary or non-existent in most body shops.

Ask first to see the organization's internal **methodology and standards** documentation. The body shop will explain the lack of such material by proclaiming "we follow our clients' standards", as if that were an organizational strength. It isn't.

Of course when a larger client organization has a sound internal methodology, the contractor is well-advised (and sometimes required) to comply with it, but

---

<sup>1</sup> Those in the Washington suburbs, specializing in government contracts are called "Beltway bandits" by insiders.

<sup>2</sup> Controversies about the tax and insurance consequences of such arrangements for all parties are well known.

that's comparatively rare. Even then, however, a mature software development firm will provide its own guidance to its own professional staff.

Next look for the organization's repository of **reusable software components**. If none exists, the body shop offers a variety of explanations. "Our projects are all so unique that . . .", was discredited decades ago and needs no rebuttal here. "Our projects are client proprietary, and they won't let us reuse components" is almost as silly. A particularly appalling one came from an executive just minutes after he had proudly asserted that his well-known Chicago-based firm is not a body shop: "Our clients pay us to *develop* software, not to *find* it."

Since it has been estimated that as much as 80% of the program code that gets written is redundant<sup>3</sup>, a library of reusable components is a precious asset. Just getting that fraction down to 50% offers a major breakthrough in every stage of software development, especially testing and maintenance.

Next, find out how the organization **evaluates the performance of its staff**. A typical body shop relies mainly on client satisfaction: "If the customer's happy with Fred, then we're happy, too." No one in the body shop firm ever examines Fred's work, much less critically evaluates it. No samples of Fred's work are kept on file in the body shop's office. If Fred claims a list of skills<sup>4</sup>, perhaps supported by vendor certification or an in-house examination, then the body shop will assign Fred to any client whose stated needs seem to match those skills.

The problem with relying on the client to evaluate Fred's work is that a naive client may have no way of distinguishing between a superbly-organized, well-documented set of programs and an unmaintainable nightmare that happens to work today. (Cynics may point out that the unmaintainable nightmare is likely to generate more future business for the body shop!)

Finally look at the organization's **professional development** (i.e. in-house *training*) program. Few body shops (virtually no brokers) provide ongoing education for their own staff. It's cheaper and easier just to engage new people in response to demand for particular skills. A professional software development firm,

on the other hand, invests almost continuously in its people. For a small firm, that may be little more than periodic after-work discussion groups; quantity is less important than regularity.

A bulletin board<sup>5</sup> announcing the schedule and topics to be covered by in-house and outside courses is a sure sign that a firm is more than a typical body shop.

Finally, look at the firm's internal **support functions**, such as:

- Set-up, administration, and trouble-shooting for its own computer systems and networks.
- Proposal preparation and packaging.
- Cataloguing and filing documentation, source code, and other materials.
- Project status and time reporting.

Many body shops view such support functions as unnecessary overhead, and leave programmers and other professionals to fend for themselves. On the other hand, the better software development firms consider them indispensable to the productivity (and to the morale) of their professional staffs, easily justifying any resulting billing-rate differential.

### ***Contracting, and staffing indicators***

The best professional software development firms take pride in their infrastructure, knowing that solid standards and reusable components give them a big competitive advantage over less enlightened firms. They, therefore, prefer **fixed-price development contracts** to hourly-rate ones for any project that has a well-defined end result<sup>6</sup>. The client, too, will prefer a contract that limits their risk of cost overrun or schedule slippage. Under a fixed price contract, individual billing rates are irrelevant to the client.

The body shop, on the other hand, thrives mainly on billable hours. Because it tends to equate skills to specific product knowledge rather than to either quality or productivity, a body shop finds it hard to place people at significantly higher-than-average rates -- if they don't believe it themselves, they can hardly persuade a naive or dubious client that greater productivity will more than offset the higher rate. Consequently they attract few really top performers to their staffs.

<sup>3</sup> That is, it has been written before somewhere, often in the same organization!

<sup>4</sup> Today, most such skills are oriented to specific software products, and focus on memorizing facts rather than mastering concepts.

<sup>5</sup> In geographically dispersed organizations, that may be replaced by Internet material.

<sup>6</sup> Professional software development firms do, of course, take on time-and-expenses projects, particularly when (a) the end result is ill-defined or (b) day-to-day project direction rests with the client.

The best software development firms often assign their people, especially those with highly specialized skills, to **multiple projects**. A staff member's participation on any one project, then, is less than full-time. That works greatly to the clients' advantage by giving them the services of top performers with specialized skills just when they're needed, people who couldn't be justified full time on that project. The contributions from such people carry a **very high multiplier**, thereby making other team members much more effective. Skills and activities that lend themselves well to that approach aren't low-level product-oriented ones, but those requiring mature judgment, such as database design, application framework architecture, complicated algorithm design, and various kinds of reviews.

The body shop, on the other hand, wants every staff member assigned full time to a single project, and that's what their clients seem to want, too. It matters little if Alice is 20 times more productive than Fred and turns out higher quality results; the broker still prefers Fred full time to Alice half time<sup>7</sup> and probably won't even offer the client that choice.

## Conclusions

If you need outside help with a software development or maintenance project, you should look for a body shop when:

- Your own organization has a competent internal Information Services organization, that is short-handed in certain skills needed in the short-term, especially in some product-oriented skill.
- You're undertaking a project that will be managed by your own people.
- Your project needs low-level detailed skills more than design or management skills.

On the other hand, you should seek a professional software development firm when:

- Your organization has no internal Information Services organization (or you lack confidence in your own staff's capability to run the project).
- You're undertaking a project that has from the start a well-defined end result.
- The project is outside the mainstream of projects your firm is used to handling internally, either by

virtue of the application area or by virtue of the proposed technology to be used.

- It is especially important to limit the risks of cost overrun or schedule slippage.
- You're just not sure exactly what you need.

## Related reading:

Conrad Weisert: "Instilling Professionalism in a Software Development Organization", Proceedings of the 1988 ACM SIGCPR Conference on the Management of Information Systems Personnel, pp 192-198.

Frederick Brooks: *The Mythical Man-Month*, 1995 (20th Anniversary Edition), Addison-Wesley, ISBN 0-201-83595-9.

---

<sup>7</sup> I get calls from time to time from brokers looking to subcontract some job, but we rarely get past the full-time/part-time and billing-rate issues. The last thing many brokers want to hear is that I'm highly *productive*!