

Drupal Support: A White Paper

May 2015

Executive Summary

Using Monarch Digital's experience in supporting and developing Drupal websites, this white paper outlines the challenges to identifying, selecting and structuring a Drupal support arrangement.

Through a case study of a small Drupal website manager, we identify selection issues, types of support, support agreements and selection criteria and, finally, what we, at Monarch Digital, feel are best practices in providing Drupal support.

Case Study

The following case study is mostly real, with some details changed to protect the anonymity of our client. This organization is a very small, for-profit company. The manager's impressions and fears are the same as those from a larger organization. They are simply more raw and unfiltered.

I like my company's Drupal site. I can quickly and easily add products, change pages, promote sales and point prospects right to what they are looking for. I can be very self-sufficient with the web site and I haven't needed any help with the site for some time.

For some time, I was seeing pinkish security warnings saying that I should update the site. Now, they have gone away. I suspect that I'm so far behind that Drupal has given up on me. I know that's important, but the original developer is long gone and I don't really know who to turn to. I asked my local hosting company, but they didn't have any ideas.

Then, it happened. I did something that made the entire site go away. When I say "go away", I mean that nothing shows up anymore. Nothing at all. I called my hosting company. They said that I need a Drupal developer and they couldn't recommend anyone.





I just googled “Drupal support”. Several didn’t answer the phone. Then, I talked to Rick at Monarch Digital. I really had no way to evaluate if Monarch Digital could help me. Rick is another Midwesterner and, as odd as that sounds, it made me feel a bit better. I felt like I was taking in my car to a new repair shop and I had no idea if I was going to be taken advantage of. I gave them all the usernames and passwords. Boy, I hope this works out.

Within hours, Monarch Digital had the site up again. They also recommended some things that would prevent this from happening again. The price was reasonable. I had them do some of the basic tasks they identified, but they also said that we were on an unsupported version of Drupal.

As I feared, I learned that I was on Drupal 5 and we were way behind. Monarch Digital gave me a quote for moving the site to Drupal 7. With the original problem behind me, I went on my merry way. I knew I had to deal with this, but not now.

January and a new budget year came around. I reached out to Monarch Digital to revisit moving to Drupal 7. Surprisingly, they came back with an estimate that was less than before and I had them convert the site.

Within a week, I had a Drupal 7 site. In addition, Monarch Digital used a theme that allowed my customers and prospects see a phone version of the site so it looks like an app.

Now, I am using the site daily and it is working great. I know I can count on Monarch Digital to back us up when (not if) we run into problems in the future. The web site is too important to us to run without a safety net.

Most people are busy running their organization and really don’t want to be responsible for all the details in operating their website. Even if the organization has its own IT department, they are usually so consumed with a myriad of projects that they are forced to put out fires, not provide proactive web site support and enhancement.

So, where does support and enhancement end and new development start again?

Types of Vendors and Support

Is a website ever “done”? If so, it’s probably not being used.

Especially in the Drupal world, there are Drupal shops that only do new development. They do not support what they have built. They move on to the next development project. They do development well and, for whatever reason, they call a site “done” and move on.

Like our case study above, the website owner progresses through stages of needs in supporting their website:

- **Emergency.** My website is down! What happened? Who can fix this thing? Who built it?

We regularly get calls like this to jump in, put out the fires, and get the site up and running like before. Like an emergency room doctor, we often find many other issues that either contributed to the problem or that will pop up as another issue in the near future.



Sometimes, we continue to be medics, being called in when the patient is near death. Other times, we move on to the next stage.

- **Support.** Like the chores on your weekly to-do list at home, there are those issues on your website that you have been trying to ignore. If you avert your eyes, perhaps they will go away. That ugly red security updates box. That page that never displayed correctly.

Like in our case study, you might trust the Drupal vendor who revived your website during an emergency. This can be the first step in trusting someone to provide ongoing support.

Once those initial annoyances are addressed, you start to allow yourself to see a number of other tasks that you now realize can be addressed.

- **Enhancement.** Now, you're starting to look past the current website functionality to new features that you've always needed. Your support vendor has had to learn your website and enough of your business to fix and support it. The same folks now have the experience to extend the functionality as well.

If the enhancement is more comprehensive, you might manage this differently than the

smaller support tasks. We'll talk more about this later.

- **Strategic.** Having built up a working relationship with your Drupal vendor, you might partner with them to identify new opportunities that are of strategic importance to your organization. In many cases, you, as the site manager, must explain business challenges and turn to the vendor for technology alternatives.

Selection Criteria

We have found that organizations and individuals select a Drupal support vendor using a wide variety of rational (and sometimes emotional) criteria.

In-house. For organizations with their own IT staff, bringing Drupal support in house can be a viable option. Very commonly, the support and enhancement of a new Drupal site is added to all of the other duties that are keeping the support team 100% busy.

We have frequently been called (back) in to support Drupal websites when the IT staff doesn't have the time to learn Drupal, to investigate the reported issues, or to actually design and implement changes.

Original Developer. This can be a very viable option if the developer is structured to provide support. Drupal shops that are organized around development projects often have difficulty dropping their longer-term tasks to immediately address urgent support issues.

During a more complex project, development organizations often shield their developers from phone calls to allow for uninterrupted working time. This flies in the face of support service levels.

Cost. It can be difficult to evaluate vendors based upon cost alone. One organization's hourly rate might be lower, but a task could easily take much longer.

Especially in support, excellent communication skills are critical. Monarch Digital has assumed support and enhancement responsibilities for a number of websites that had previously been supported by non-English speaking overseas outsourcing companies. In several cases, it was obvious that the developer misunderstood the requirements and programmed something completely different than what the site owner needed.

Just like a handyman who doesn't like to do plumbing, some Drupal shops will provide exceptionally high bids on support. If the client accepts the bid, they will do support even though they don't like to.



Geographic. I've known many website owners who prefer to do business with someone local as an additional accountability, trust and reliability factor. This can be valid. However, keep in mind that online conferencing and good vendor communications can even trump a local vendor who is not as responsive as you need.

Industry Experience. Personally, I feel that industry experience is more important during initial website development than in the support phase. When you are supporting existing functionality, you're already working inside the structure created during development. As the Drupal support vendor gets more experience with your website and moves from support to enhancement and beyond, industry experience pales compared to project-specific familiarity.

References. Good references are always valuable. However, be sure to ask questions specifically about support issues: response times, quickly understanding new, complex issues, feedback and communications, etc.

Communications, Trust and Service. In everything listed above, website owners are really trying to gauge how well they can communicate their issues, and trust that the vendor will address the issue promptly and correctly.

Drupal shops that do support well are extremely customer-focused. They answer the phone. They will drop everything to address an emergency. They have the experience to (gently) direct you away from remedies that will hurt you in the future.

Structure of Support Agreements

There are many ways that Requests for Proposals (RFP's) do not work for web development projects. In most cases, RFP's don't work well for support either. In one case, I had a client present us with a list of specific projects they needed to have completed. We talked through the projects and examined their website to provide them with estimates on their specific needs. As we did further research, we found that we inherited spaghetti code. Then, the more items we addressed, the more got added to our list.

There are many ways that Drupal shops structure their support agreements.

- **Monthly support agreement over a fixed term.** This is one of the most popular with Drupal shops, not necessarily that popular with clients. The Drupal shop would charge a monthly amount over a commitment period of, say, a year. The client receives X hours per month for that amount.

I have found that some Drupal vendors enforce a monthly "use-it-or-lose-it policy". If you don't use your hours this month, you've paid for hours you don't use.



In defense of companies that like to work this way, they say that they will dedicate X hours of a developer to you every month. Whether you use the developer or not, they have dedicated a portion of their time to you.

Monarch Digital uses a variation of this support agreement. We provide a fixed budget amount of time and dollars monthly. We don't usually have any term. It's purely month to month. If the hours are not used in one month, they carry over to the future.

Used in this way, a monthly support agreement helps in budgeting support dollars and prioritizing projects within a specified number of hours monthly.

- **Price over a fixed term.** In this scenario, the Drupal shop invoices the client a fixed amount of money for a fixed amount of hours over a time period, say one year. The Drupal shop likes to get paid up front. (Don't we all?)

Personally, I don't like this structure. There is little incentive for the vendor to respond quickly, to perform in a timely fashion, or to perform at all.

- **Upfront payment.** As a client, you have a couple of small projects. Neither of you have done business in the past. The Drupal shop provides a client with an estimate. The client pays up front as they anticipate that this is all they will need of this vendor. The Drupal shop requires an upfront payment to assure that they will be paid for their work.

Monarch Digital has used this structure for small (less than three hours) one-time projects. Frankly, it is not very comfortable for either party. The client is paying for something they don't have yet. The Drupal vendor knows this client relationship isn't going anywhere.

- **Pay as you go.** Interestingly enough, Drupal shops don't propose this very often. Personally, I think this is the most sustainable for a long term relationship, given that the two parties mutually trust the respect one another.

Basically, the client pays for the time that the Drupal shop actually spends on the site.

In a less than ideal situation, a client could feel that the Drupal shop is incentivized to record more hours than they actually spend. A Drupal shop could feel that they could be left holding a bad receivable for as many as one or two months of support if the client decided not to pay. (Unfortunately, Monarch Digital has been in this situation.)

Best Practices

Over the years, Monarch Digital has adopted the best of all of the experiences and alternatives above. I'm sure there are other structures that work, but we have gotten a great deal of positive feedback on how we do support compared to other alternatives.

Small emergency projects. Monarch Digital does do small, emergency projects for new clients, provided we have manpower available. We prefer to help small nonprofits or small governments like this.

Monthly support contract. We have come up with a hybrid of the pay-as-you-go model above. After initial conversations, the client signs a monthly support contract:

- The client identifies the task(s) to Monarch Digital.
- Monarch Digital then provides an estimate to the client.
- If the client accepts the estimate, Monarch Digital works on the project and delivers it to the client.
- If Monarch Digital spends any time on the client's project in the previous week, Monarch sends a detailed weekly time report to the client. The client is fully informed on what time is being spent on the project and can better manage their budget.
- At the end of the month, Monarch Digital sends a detailed time report for the month along with an invoice.

Support Configuration

Monarch Digital prefers to set up the following configuration to provide the best support for our customers. For smaller clients or for clients in an emergency situation, we may not create all of this structure to save time and/or money.

Staging site. You just don't want to try untested changes on the production website that the entire world sees. (Unless it is an absolute emergency.) Monarch Digital would initially clone the production code, files and database. This site would effectively be a sandbox for changes before they are applied to production.

Local site. Monarch Digital's developer also clones the production site on their local machine as a development environment. Without this, the developers could implement untested changes that might completely bring down a site.





Release control. As part of working on your site, Monarch Digital makes changes to your code. A release control system (like git) will store every change made to your code. At any time, we could back off a change to your code, reverting changes to a previous version. Release control also allows us to easily review code changes and to pull the same version of code to production, staging or local versions of your site.

Drush. Drush is a Drupal command line utility that improves the productivity of Drupal developers. With a single command, developers can update modules, clear cache, change passwords and much, much more.

Training. The entire purpose of a CMS is to allow non-technical content managers to be self-reliant. Only call in developers when needed. Often, content managers have been working without developer backup. We can coordinate with and train client content managers to make their upgrade and testing workflow as efficient as possible in working with us.

Backups. For longer term customers, we have configured weekly backups of their websites and databases, providing the ability to recreate them from weeks in the past. For some transactional, e-commerce sites, Monarch Digital has kept archive copies of the websites for a year or more.

Conclusion

Although website development and support are different animals, they also can transition from support to new development and back. Using our experience in both realms, we wanted to give some structure to procuring and managing Drupal support along with that transition from large support project to new development (or re-development).

At Monarch Digital, we support many types of Drupal sites. Feel free to call or [email](#) for additional information.

Reference posts:

- Upgrading: [So you want to upgrade your Drupal site](#)
- Cost: [Comparing your Drupal support alternatives](#)
- Release control and a test site: [Does our site really need release control and a test site?](#)
- Backups: [Could you restore your web site?](#)