Welcome to the CODE Ohana!

Markus Egger

Founder, President, and Chief Software Architect

Preface

I always knew that I wanted to start my own company. I did so back in 1993 when EPS Software incorporated. I was 19 years old at the time. EPS Software has now been in existence considerably more than half my life. And what a ride it has been!

I have always had a passion for computers and software, and, perhaps more unique for a geek, a passion for business. Thus creating my own company, even though entrepreneurship like this is not common in my native country Austria, was kind of a given for me. There were quite a number of hoops to jump through (especially since getting business licenses for someone "that young" was not easy at the time), but it all worked out. I am glad it did.

There is no such thing as "being self-employed" in Austria, hence EPS has always been a properly incorporated company from day one. However, a few years later, when it became clear that was cost prohibitive to grow a business in Europe, I "outsourced to America". It simply made more sense to do so, especially since EPS already had many customers in the US (such as Microsoft, Qualcomm, and others). After jumping through more hoops, EPS Software Corp. opened its doors in Houston is 1999, and we are still here today. (And BTW, there still also is an "EPS Austria", although this is a new company we re-established a number of years ago and not a direct continuation of the original company).

Throughout the history of the company, a lot has changed. Technologies have come and gone several times over. We have even adjusted and tuned our services and our approach to business in general. But at core, our purpose has always remained the same - **We help people and companies build better software!** The specifics of this may change, but this core mantra will always remain the same. Everything we do revolves around this idea. This is our purpose!

How to Use This Book

This book is meant to provide guidance and to share ideas. This book does not cover the details of your compensation package or contract structure. There is other documentation for that. This book is meant to share our core ideas and values in the hopes that you gain an understanding of “what makes us tick”. We are not a large enterprise with rigid hierarchical structures.

Instead, we strive to keep things simple and to enable everyone to grow into the role(s) they envision and like best. We like working with people who can work independently and intelligently once given a goal and basic direction. This book aims to provide at least a piece of that puzzle. It aims to keep you from panicking, now that you are here.

Aloha!

Welcome! You are now a member of the CODE Ohana.

“Ohana” is Hawaiian and means “extended family”. This perfectly describes how we view everyone who is part of our organization. We may live and work in different places. We may have different backgrounds. We may be using different technologies and work on different projects. We may even work in and for completely different parts of the business. At the end of the day, we are one big ohana. We get together on occasion. We interact with each other. We are there for each other. Whether you are reading this because you have just become a full-time employee, a part-timer, an intern, a contractor, a contributor, or any other kind of relationship with EPS, you are now – to paraphrase the Godfather – “part of the family”. Or, in more geeky terms, you have been assimilated. Congratulations, and welcome!

Like any other company and organization, we have our own unique approach, our own philosophy, and our quirks. Perhaps more so than others. Some of it may take some getting used to. In many ways, no matter how tight your official relationship with the company is, and no matter whether you are an employee or just a remote contributor, our structure provides an enormous amount of freedom and the ability to structure your own approach. That can be liberating, and it can be intimidating. It is both one of our greatest strengths as well as our greatest weakness. It can take some getting used to, especially if you are used to more rigid corporate structures. This book was created to make you feel as comfortable and as welcome as possible, and to allow you to ease into the place you choose for yourself.

Company Facts That Matter

EPS is completely self-funded. We have never brought in outside investment. This is an incredibly important aspect in understanding many of the goings-on and the decisions made in the company. For us, every decision matters, and every decision can be (and usually is) significant. A large project gone wrong can have dire consequences, so we do our darndest to make sure that doesn't happen.

EPS has been around far longer than most of our competitors. This means that we have made many decisions that worked out. We have also made many bad decisions I am sure, but we have been able to make more good ones, which has kept us around much longer than others. This is a fact we are proud of, even if the going wasn't always easy.

We are also a company that is dependent on outside influences. The overall health of the economy has a huge impact on our operations, after all, our customers can always delay an IT investment by a year or two. How well partner companies like Microsoft are doing is significant, as was more than apparent during the "dark days of Windows 8" compared to more recent times. The technologies we choose to use and support have a huge impact on our overall wellbeing.

We are however in the fortunate position to say that we never had to leave a bill unpaid, and we never had to lay-off an employee for financial reasons. We may have parted ways with people that were not a good match, but it was never because we had to cut salaries. We did go through extended periods of times where leadership had to take serious cuts, Markus and Ellen have gone without paychecks for months at a time, and staff have realized fewer bonuses and raises when we had to tighten our belts. But even in the toughest times of .com bubbles bursting and the economy collapsing, we have been able to carry on and – as cliché as this may sound – emerge in a better position than before. It's not all puppies and rainbows (or unicorns, as the case may be), and it never is, but we are still proud of our history.

We may have never grown fast enough to make the INC 500, although it was close at one point, but we also avoided a belly flop. It would be an understatement to say that we are proud of that.

Company Overview

Markus Egger founded EPS Software in 1993. Although only 19 years old at the time, Markus was already well known in the developer community. In addition to doing development work, Markus is an international author and speaker. Markus has also been a contractor at Microsoft in Redmond “back in the day when working at Microsoft was like Disneyland for adults”, as he likes to say. He worked as a contractor to the Visual Studio team, with his own window office just around the corner from Bill Gates’ office. Some of the remnants of some of Markus’ early code can still be found in Visual Studio today.

Markus also has been a Microsoft MVP since 1995, which makes him one of the two longest running MVPs, and currently the Microsoft Regional Director (“RD”) for the Central USA Region. Markus is also part of various insider networks, such as the “ASP.NET Insiders”, and various SDR (“Strategic Design Review”) Teams such as C#, UI tech, and others.

EPS started with the same core business idea as is still at the core today: Consulting and Custom Software. In addition, other aspects of the “Helping Companies Build Better Software” mantra have led to different services and offerings. Training has been a long-term offering, as has community involvement, and event presentations. Products have played an important role throughout EPS’ existence, although not for the product’s sake but as tools to build credibility and trust with our clients. Then there is CODE Magazine, that has now been around for quite some time, much longer than just about any other independent software magazine.

By now, you may be wondering why there is a certain Hawaiian theme to this introduction. This is because both Markus and Rick (the co-founder of CODE Magazine) live in Hawaii.

What is the "CODE" Thing All About?

So is it CODE, or is it EPS? It is confusing! The legal name of the company is "EPS Software". In the US, it is "EPS Software Corp.", in Austria; it is "EPS Software GmbH". Either way, "EPS" is the official company name (it originally stood for "Egger Programming Services"). So where did the whole "CODE Thing" come from?

Well, we decided to launch a magazine. And magazines need names. When we started the magazine, developing "components" had just become a real thing (in the COM and Windows DNA days), and developing language-independent software, presumably using components, was a major topic of the magazine. Therefore, we decided to call the magazine "Component Developer Magazine", or short, "CoDe Magazine". In fact, we always liked CODE better, but we needed to stick with Component Developer for a while for copyright and trademark reasons. Larry Flint of Hustler Magazine fame owned his own magazine called "Code Magazine", which was an unfortunate coincidence... but it resolved itself when his magazine folded after just a few issues. It did however lead to some embarrassing moments for readers when they directed their browsers to codemagazine.com, which redirected into Larry's "other sites".

Long story short: We ended up with a magazine called CODE Magazine, and over time, it simply made sense to also brand our training as "CODE Training" since it was the training arm of the magazine. Once we travelled down that path, and considering that CODE Magazine had become a well-trusted brand, it simply made sense to brand all our other offerings with the "CODE" moniker.

It simply made sense to use the CODE branding for all our divisions. The CODE brand is owned by a separate company called Markus Egger Inc., which is the custodian of these brands and licenses them to EPS as well as to other entities internationally. It leads to a little bit of confusion and to some awkward web site and email domains, but overall the CODE brand works out pretty well for us.

CODE
An EPS Brand

Your First Day

So now that you are here and “assimilated”, how do you get started? We know it can be quite overwhelming.

The answer to that question will vary greatly depending on the capacity in which you have joined us. If you have simply become a member of DevNet, there are no specific tasks, expectations, or obligations. You should have gotten your welcome package, and you probably have access to the **devnet@codemag.com** email list, and you are ready to go. Lean back, fasten your seatbelt, and enjoy the ride! :-) The only thing we ask is that you make sure we have your up-to-date information in case anything changes (such as email addresses, physical addresses, technical expertise,...) which allows us to integrate you better and make sure the projects that are of interest to you are actually flowing your way.

If you are an employee or contractor, a more active role is suggested. By now you should have gotten your account information to log into our infrastructure, and you probably signed all your paperwork. EPS used to be a very traditional brick & mortar operation, and if you are physically in our offices, your co-workers will show you around. However, chances are that you are an “off-site resource” who comes to the office only occasionally, if ever, in which case make sure that you can log in remotely and that you have all our standard tools and infrastructure set up. You will be receiving separate documentation about this process.

In addition, it never hurts to familiarize yourself with all our different divisions. Poke around our web site (codemag.com) and check out all the different things that are going on. Take a look at the consulting and training options we offer, take note of upcoming events, and, of course, check out the magazine site. For all developers, particular attention should be paid to the CODE Framework area, which lists many links to technical resources. We don’t have a formal “required reading” list, but the introductory CODE Framework articles are be highly recommended. If you are a contractor looking to be assigned to projects, then it certainly helps a whole lot to be familiar with the technologies and processes we use (hint, hint! :-).

Also note that we have an internal training program we refer to as “DEPs” (which stands for “Developer Education Program”). You can read more about DEPs below, but they are important to mention here, because we record all our training events. Hence going back and watching existing recordings of prior training sessions is a great way to familiarize yourself with the things you need to know.

Beyond that, make sure you help other people in the organization to do their jobs. Like every other company, we have a few chores that need to be taken care of every week. This includes doing your time sheets (so we can create client invoices and analyze our expenses - important tasks from an overall company point of view) and if you are an employee, sending around your “3 Wins” every Monday. Not doing these tasks will quickly eliminate you from the friends-list of those of us who require these items to get their own jobs done. If you do those timely on the other hand, you will find that we do not need to bother you with a lot of micro management.

That’s just about it! Most likely, your manager (or mentor) will already have a few tasks for you to get started on.

Keeping the Hale in Shape

“Hale” is another Hawaiian word. It means “house”.

We are talking about housekeeping here. There are a few items that must be done that are chores for all of us, but are simply needed to operate the business.

ClockWork Timesheets

One of them is the need to keep track of our time. We need to do this for two reasons. For one, we need time sheets to know how much to bill our customers. Everyone working on billable tasks has to enter those tasks into ClockWork with appropriate start and end times and a description. Note that the customer can see that description, so please be aware of this, and keep it professional. This is not the place for silly jokes :-). Secondly, we need the time logs for our own analysis. Think about it like this: We can either do a lot of micro-management, and be on everyone’s case all the time and constantly look over your shoulder, or we can rely on time logs and get the information we need for analytics that way. I would say it is much better to do time sheets, isn’t it? I thought so.

Be aware that we need your time sheets first thing every Monday morning. This goes for everyone! You can make Colleen’s job much easier by submitting your time sheets on time. Think about it: We have a bottleneck there, since many tasks, like invoicing, cannot happen until we have all the time sheets. So just one missing timesheet is really a problem that keeps people from doing their jobs. So don’t be surprised if Colleen turns into Pele, the goddess of fire, and comes down on you like a ton of lava rocks, if you do not turn them in. Trust me. You do not want to mess with Pele!

Timesheets are required from everyone who expects to receive money from the company, employees and contractors. The only people exempt from this are *CODE Magazine* authors.

Three Wins

Another long-time management staple at EPS is the concept of “three wins”. The idea here is that everyone should think of three things they want to achieve in the coming week. Three is a nice number, because it keeps things simple and achievable. So every Monday morning, we want you to send an email with your three wins for the week. We also want to report on whether or not you achieved what you had aimed for in the previous week.

Again, doing the “three wins” allows us to get away from micro-management and instead allows people to self-manage. Management stays informed of your goals for the week, which allows us to re-prioritize if need be. It also gives us an idea as to whether we achieve what we set out do to, and so forth.

“Three wins” are required only from EPS employees.

Paddling the Outrigger

EPS does many things, but what keeps our boat afloat? Simply put - writing software! Everything we do pretty much revolves around that, and the task of writing custom software and providing consulting services for our customers is our core business.

Not only is this a core part of our business, but it is also the most difficult thing we do. Each project is super-important to us, because software projects tend to be large, and messing up even a single project has very dire consequences.

The Development Process

We have quite a history in development processes. Markus has a lot of background in processes such as Waterfall, the Unified Process, and Agile. In fact, the first time Microsoft hired Markus they put him in charge of Rational Rose integration into Visual Studio. Rational Software is the inventor of UML and the Unified Process and was later purchased by Microsoft. So with all this background in various processes, which one do we use for our projects?

There is no simple “one size fits all” answer. Instead, we adjust our process to customer needs as well as the technical characteristics of each project. However, it is fair to say that we always aim for a very simple and lean process based on Agile. We find “Less is more” is good guidance.

There always is a question about which tools we use for requirements analysis, work item tracking, and more. Some of that is standardized through technologies and platforms such as TFS, although the approach needs to be adjusted to the needs of each customer. For others, we like to give teams the freedom to use the tools that work best for them.

However, whenever we add new tools to the mix, we need to discuss it and make the decision together. It is important that such assets are always accessible and can be maintained and accessed over time. We do not want to switch to some new tool on a whim just “for the heck of it” and then perhaps face challenges in accessing that content a year or two later.

Asset Management

Source and Asset Control seems to be a hot topic, in many ways to a surprising extent. The way we look at it is similar to our development process - keep things simple. Over time, we have tried a number of different approaches, ranging from Visual SourceSafe to TFS, and more recently, things such as GitHub and Mercurial. At the end of the day, we seem to always come around to using Team Foundation Server in some incarnation, often using Git. That should be your default assumption.

Note that EPS has a history of long-term maintenance and reuse of code. Probably far more so than other companies. There is a lot of value in having our assets in a shared store that goes back decades and includes all projects we have been working on. For us, this single shared store, is Team Foundation Server.

Our Pow Wows

Like any other software company, we tend to have meetings. Maybe not as many as others, but still quite a few. We meet amongst each other, have team scrums, meet with customers. We are constantly evolving our meeting strategies in an effort to keep meeting times to a minimum, yet still allow a geographically disjointed team to act as a whole.

Our current meeting strategy is a combination of Skype, GoToMeeting, and MS Teams. This may have evolved by the time you read this, so ask around.

One staple of our meetings is that we record almost all of them. We use built-in recording features (such as GoToMeeting's recording feature), or an external tool (such as Camtasia) to record meetings. Most likely, as the newcomer, you won't have to worry about recording it, since it is usually the meeting organizer who does the recording, however, once you get to a point where you call your own meetings, make sure you do not forget to record them. The recording gets uploaded to Screencast.com, filed away neatly into the appropriate folder.

Remember Pele, the goddess of late timesheets? Well, we also have a god of forgotten recordings. His name is Kūkaʻilimoku, or Ku in short. He is even scarier. Look him up under "human sacrifices" :-)

Presenting Progress to the Customer

For those who are working on customer-facing projects, please make sure you consult with your manager about how exactly to communicate with the customer for that project. There is however, one aspect that remains the same for every project - we provide progress updates to customers in the form of short videos. Typically, these videos are a few minutes long and show off all the new things that have been done and why.

There is a very important reason for these videos. They document the progress of the project, and they document why certain things were done. We have learned the hard way that having this kind of backup information is very, very valuable for a variety of reasons. For instance, it can be valuable when we need to go back to see what we decided in a certain meeting or iteration. It can be valuable when we need to see what was accomplished. It can be super valuable even for you specifically, in case someone gives you a hard time about what you programmed and why.

This is something we do always and for every project. There is no excuse for not doing it. Remember Ku, the God of War? Boy, does he love his progress update videos ;-)

Doing What's Right for the Customer

At EPS, we have a simple philosophy in terms of our relationship with our customers - whenever possible, we strive to put what is right for the customer first. We simply ask ourselves "what would be the best possible scenario from the customer's point-of-view that would make the customer love us and our services and/or products?" We then try to implement that as much as possible.

Let me provide an example: Our magazine subscribers often have the need to change their delivery address so they can still get their magazines after they move to a new home. In the past, they had to call or email us to change their address. This had to do with the fact that if a customer were to move to a different country, the mailing expenses changed, and figuring out what the new pricing should be, was a complex process on our end. Therefore, we did not let them change the address on our web site. But then we changed our mind, and we said "it would clearly be the best for the customer to change their own address". We now support this on our web site. We still have the pricing issue, but that is something we deal with, rather than inconveniencing the customer. We now manually handle this, and as it turns out, scenarios where complex price re-calculations apply are rare. So much so, that we can pretty much ignore them. Spending even an hour or two on this would be more expensive than the potential increase in mailing expenses for all customers combined.

However, note that there are common sense limits to this. One could argue that clearly, it would be the best for the customers to get their subscriptions free. But from a business point of view this simply isn't feasible, and nobody in their right mind would argue for that.

The same applies for the software business. Scenarios are more complex, so we have to talk about them on a case-by-case basis. But ultimately, we want to apply the same principle. Let's do what's right for the customer, if at all possible!

Tools of the Trade

An interesting aspect of software development is the question of what tools, technologies, and platforms to use. Historically we are a Microsoft shop, but that does not mean we are limited to Microsoft tools. In today's world, we use a range of tools and platforms, including Windows and Linux, different Cloud platforms, different development tools (such as the Visual Studio family of products, as well as others), programming languages (C#, JavaScript, Java, Swift, and many more), and so on.

As you can see, we are open to a lot. Where things get tricky is when we add new tools and new frameworks. Especially in web development, there is always a "framework of the week" and a new library or tool. We are open about adding new ones, but it needs to be a well thought out decision. If you have something new you would like to add, let's talk about it! We need to make sure it fits into the overall strategy. We need to make sure we understand the long-term implications. We need to make sure we understand what it means for the customer and whether the customer even allows it. We need to understand the dependencies. We need to make sure we can still guarantee the quality of the overall product as well as the newly added component, since we are legally responsible for both.

So as you can see, there is a lot that goes into decisions like this. We also can't always chase the latest and greatest thing in production projects we work on for clients. Imagine if every single person on every project was to simply go off and add new things in an uncoordinated way! Software development is already crazy enough without this! :-)

Nevertheless, adding new technologies in an orderly fashion is an important aspect of what we do. Let's do it in an orderly fashion and include other people in the process. Perhaps the next DEP can even be about that cool new thing you found!

Not Invented Here… or is it?

In the software industry, I often hear people complain about the "not-invented-here syndrome". What they mean by that is that developers too often want to re-invent the wheel rather than using technology that is already available. I share this sentiment to some degree. Why re-invent when what we need is already there? However, I feel there are limits to this. After all, we are not just implementers, sitting in a large corporate IT department.

I definitely see our organization as engineers and inventors who think up and provide new solutions to improve the overall development landscape. Historically speaking, we have been very successful with that. While I certainly do not want to re-invent the wheel, I do want us to question whether an existing wheel is what we really need. And if we discover that we need wings instead, we need to evaluate what kinds of wings they need to be. You do not just bolt 747 wings on your contraption, because "they surely must be good enough". We want wings that provide the best-case scenario for our specific needs, and that often means developing new wings. I expect our developers to have the skills to do so. Well, the software equivalent anyway. :-)

What we really need is a good way to balance the use of existing technology and the invention of new components. We need to carefully evaluate whether something that is already available will allow us to operate at the level of quality we require. We need to make sure we fully understand the implications of bringing in an existing component or technology. Too often, I see developers using something that "kind-of matches", without a comprehensive understanding of the implications and dependencies. I do not consider that acceptable.

How We Bill Our Customers

Most of you do not have to worry about how exactly we bill our customers, but we would still like to provide some background information about the process.

In almost all cases, we bill our customers based on the actual hours we work. In other words: We sell our services, rather than some kind of finished product. This also means that we do not do fixed-price projects. Or, at least, the cases where we do are few and far between.

In terms of the actual amount we bill for each person, it depends on what technical "level" we consider the person to be. We have four different skill-levels, from 1 (lowest... typically testers or junior developers), up to 4 (renowned experts, authors, and speakers at major events). Most developers fall into levels 2 (a professional with considerable expertise) and 3 (a senior developer with leadership skills, who is usually also a published author and speaker).

A Word about Contracting

Some of you reading this will be full-time employees. If so, you can skip to the next section. For those of you who are contracting with EPS, please make sure you understand what contracting means. We have additional documentation for those who are new to such a setup, so please refer to that for more information. However, the short version is that you are a vendor. You provide services or goods that EPS buys from you. You are not an employee. You need to have your all own tools including computers and software licenses, you need to pay your own taxes, you set your own hours, and so forth.

We have a policy of inviting many of our contractors to various EPS initiatives. We have training, events and other initiatives that are open to anyone, ranging from EPS employees to contractors, and even simply members of the development community that have no other business relationship with EPS. While we are happy to extend those benefits indiscriminately, there is no guarantee we will continue to provide them or that we will provide them in the same fashion and to the same group of people.

What We are Not Good At

We try our best, but clearly, we are not good at everything. As is often the case, our biggest weakness is also one of our strengths - we are a geographically disjointed group of people with different backgrounds, different experiences, and different technical expertise. We are also not good at micro-managing people. Many would not consider that a negative. :-) However, we do have to acknowledge that we are not good at "managing people into the best they can be". We simply rely quite a bit on people being able to organize themselves within the boundary conditions we define. We also often rely on team members bringing issues to our attention because due to the virtual nature of our teams, issues that are obvious to some, are often completely unknown to management.

We acknowledge these weaknesses and are steadily working to improve them, but we rely on each individual to help us work towards improvement.

Another problematic area is that we are simply always super busy. We are not a large enterprise that can set aside a lot of resources for overhead tasks. One can argue that such is the reality of life, even for many large organizations. Yet it certainly is an area we are aware of and try to improve, although it remains an elusive goal.

Talk to the Big Kahuna

Getting used to a new environment is never easy. There are all the little things that one simply absorbs over time but that are foreign when new. Don't be shy to ask questions! We want everyone at EPS to have a person that is their primary go-to point when it comes to question, no matter whether they are technical or otherwise. Think of this person as a mentor. If you do not know who your mentor is, simply ask Mike, Ellen, or Markus.

There are always questions that can't be answered that easily. Sometimes the Menehune are in the detail, as we say in Hawai'i. (Menehune are like Hawaiian Gremlins). If you can't get your questions answered, make sure you talk to Markus, Mike, or Ellen. While all three tend to be perpetually busy, which is why it is easier to talk to your mentor first, they are ultimately there to make sure you can work productively and do not encounter roadblocks. Send an email to them whenever you feel you encounter a roadblock, or when you feel you would like to give feedback of some sort. They will always have an open ear.

Community Involvement

You may already be aware that EPS and CODE have always pushed involvement in the developer community and that we have provided support in any way that is feasible. We encourage you to participate in this spirit.

If there is a user group or a conference in your area, we encourage you to get involved. If you have no contacts with anyone in your area, we can help you set it up. You wouldn't believe how much of an ice breaker a free box of CODE Magazines can be ;-). If you end up representing EPS at the event, there may even be funds for sponsorships or similar activities. If that seems appropriate for your user group, talk to Ellen about it.

In addition, if you are interested in presenting at user groups and conferences, we can help you with these types of things. We have many connections to user group and conference organizers. We can also coach you into becoming a better presenter and help you get accepted as a speaker at various events.

MVPs and RDs

EPS has always been involved with the Microsoft Regional Director and Microsoft MVP (Most Valuable Professional) programs. Markus is an RD, and he is one of the two remaining, longest running MVPs. (Markus has been an MVP since 1995... so more than half his life, almost as long as EPS has been in existence).

If you are interested in becoming an MVP, EPS may be able to help you. There is no guarantee of course, Microsoft has to accept you into the MVP program, and that is a long and uphill battle, but it is easier with our support.

Our Own Events

We have always put on our own events. This ranges from smaller events, such as user groups and training classes, all the way up to relatively large conferences. Currently, we are mostly focusing on various training initiatives (often a "Day of..." one-day events) and State of .NET events. If you are interested in participating in any of these events, please don't hesitate to let us know!

Spreading the Aloha

We are a business, and businesses need to sell themselves and their products. That is what we do. That is our purpose. To achieve this, our main strategy has always been to make contacts in the community and show off our abilities by providing free information, doing presentations, writing free tools and products, and so on. It works out great for everyone. We think of it as "spreading the Aloha". Many people get free stuff and information from us, no strings attached, and every so often, one of them notices that we do indeed know what we are talking about, and thus decides to hire us. Everything we do for sales and marketing can be traced back to this concept. We think it is much better than a conventional sales and marketing approach.

Of course, there is a lot of effort and expense that goes into this method. Anything from CODE Magazine to speaking at conferences and supporting the community falls under this umbrella. We want everyone in the CODE Ohana to be part of this effort. Some may be more interested in this than others, but nobody escapes this aloha spirit.

In practical terms, this means that everyone is part of our community efforts. We want you to alert us of community efforts you feel we should be involved in. We would like you to go to user groups and represent CODE. We want you to give away free CODE Magazines. We want you to collect business cards. We want you to give presentations (if you enjoy doing that). And so on, and so on.

Ultimately, of course, we want you to collect leads and talk intelligently to customers about new projects. At the end of the day, finding projects is what keeps us going and what keeps you employed. Besides, for those who drum up business, there is extra compensation in the form of commission payments.

All of this we officially refer to as our "Evangelism Program". We have meetings about this every so often. If you want to be part of this in a deep fashion, please let us know, and we are more than happy to include you in our next meeting.

Lunch with CODE

We have a number of standard initiatives we use to spread the word about our services. As mentioned above, these include our State of .NET and CODE training events. Another is our “Lunch with CODE” initiative where we go on-site with potential customers and do a lunch presentation. We even bring a free lunch for them. A pre-condition is that the company we do this for has a sizable group of people (10 or more) that are interested in the presentation. We then let the customer pick a topic they are interested in and we go on-site and do the presentation. The customer gets a free training session for an hour or two, and we get in contact and establish a relationship with a potentially interesting customer.

Free Hour of Consulting

Another standard initiative is our free hour of consulting. It is a natural first step of engagement with a potential customer often whom we met at an event or who has contacted us after visiting our website. We offer to meet with the potential customer for an hour (or so), either online or in person, and discuss their specific technical. Again, this is a great way to get in contact with a customer and establish a relationship.

Contact us today for a FREE 1-hour software consultation

info@codemag.com

Writing
(for CODE Magazine and others)

Have you written for CODE Magazine yet? Perhaps you are receiving this book because you are an author. Otherwise, consider writing for CODE Magazine! If you can't think of something to write about, let's talk! If you have never written an article and are nervous about it, don't be! We have great guides for writing articles, and we have great editors (Rod and Melanie) who will make sure that your article makes sense and reads well.

For more information about writing for CODE Magazine, see our writer's guide at **www.codemag.com/write**.

Also, if you are part of the CODE Ohana, that doesn't mean you can only write for CODE. We would be thrilled to have you write for any of our competitors. In fact, if you write an article and you represent yourself as a CODE/EPS person, we will even award you a $100 Amazon gift certificate (and of course, you get to keep whatever honorarium you receive for writing the article). However, it has to be a "real" magazine (no, your own blog doesn't count for this :-) and it has to be a "real" article, rather than just a small blurb, or something of that nature.

Oh, and talking about blogging: If you have a blog of any significance, and you indicate your blog's affiliation with EPS/CODE, let's chat about this too! Perhaps there are ways we can leverage that and at the same time give you more exposure. Besides, there might be certain additional benefits with this as well.

DevNet

Maybe you are simply getting this book because you just joined DevNet. If so, not everything in this book applies to you, since you may not be doing any contracting or employee work for us. Still, welcome to the CODE Ohana!

The goal of DevNet is to create a community of professional developers who want to be part of the CODE Ohana, and both contribute and benefit. You will have access to our DevNet email alias (**devnet@codemag.com**), which allows you to fire questions to our group, so you always have the whole CODE Ohana behind you, as a kind of safety net. We may not be able to answer every single question, but often someone in our group can, and when not, sometimes we elevate it to someone like a Microsoft product group. And of course, we encourage you to actively participate and answer questions when they arise.

In addition to our email alias, we also have semi-regular internal training sessions called "DEPs" (see section below). Everyone on the DevNet alias is welcome to attend those. This also goes for most of our customer-facing training events. As long as we have room and there is no hard expense associated with it for us, you are welcome to attend all of those free.

Developer Education Program (DEP)

Being a software developer means that you are stuck in a continuous learning cycle. We want to support all our developers in this process, and we do so through our DEP program. Let us be clear on what this means - we are not a school. You will not get all our training needs satisfied by attending DEPs, and you are responsible for keeping up with the latest technologies. But DEPs are interesting training events that can be attended online and also recorded. They usually take an hour and a half (or so), and discuss a certain topic the presenter volunteered to present on. Sometimes DEPs are a conference session we are practicing to present to a wider audience. Sometimes they may be a very informal where someone shows off something new and cool they have been experimenting with.

In addition to attending DEPs, we also encourage you to actively participate in them by presenting yourself. If you have never presented, don't worry, we can help you with that! Besides, it is much easier to present to our ohana than to do it to a group of strangers. You can even team up with someone else to do a DEP together, which is a great way to get started.

CODE CONSULTING

A Closer Look at our Business

When we look at the various divisions within EPS, things can quickly look overwhelming, but it really isn't. Everything EPS does revolves around **helping companies build better software.** We have the following divisions and brands, all of which have their areas on our web site (codemag.com), which we encourage you to become acquainted with in detail.

CODE Consulting is the consulting arm of EPS Software Corp. and our primary revenue stream. Our primary focus is building custom software applications on any scale, from small consultations, to start to finish custom software projects that last for years and involve lage teams of people. We provide software development, mentoring, project management and architectural expertise. We use a variety of tools and technologies such as .NET, iOS, Android, Web, Mobile, Cloud, Services, various Databases, and more. We help our clients build applications better and faster, and help them improve quality, maintainability and reduce their time to market, reducing the risks that are inherent in every software development project.

- We maintain a core group of about 15-25 people
- We have an extended group of key experts to draw from
- We do projects ranging from just a few hours of mentoring to multi-year, multi-million-dollar projects with a dozen developers or more
- We often end up managing a customer's team members as well
- Our projects often include direct client engagement on legacy, existing, or new applications
- We rely on heavy repeat and referral business
- We work in all vertical markets including oil and gas, finance, healthcare, etc

As Markus always likes to tell our customers: *"We are big enough to handle large projects, but small enough for every project to matter"*.

CODE MAGAZINE

CODE Magazine is... well... our magazine.

You probably already knew that. :-) You probably already knew that we aim to provide in-depth content in an independent fashion. We are one of a few, if not the only, magazine that is independent and is still published as a print publication. Of course, our digital formats are of importance to us as well, but we have no intentions to stop the print version. Quite the contrary! Print has gone up in significance in recent times.

Think of it this way: How many magazines have you seen go “online only” and now they are a blog at best, or have disappeared entirely? Quite a few. We like to be a print publication, because it means you are a “real magazine”. Besides, many readers like having a printed version of a magazine, and there simply aren’t too many alternatives left. This doesn’t mean the digital format isn’t important, it simply means we will push both.

Magazines aren’t money-makers of course, and we are no exception. *CODE Magazine* is finances out of our marketing budget. However, while we do see the overall magazine effort as a marketing initiative, the actual content of the magazine is NOT driven by EPS marketing. We do not publish content that talks about how great EPS is, instead, we focus 100% on producing great content people want to read. Having a magazine with great content that people like serves us far better in gaining status and respectability than any kind of marketing speak ever would.

CODE
Slay the JavaScript Dragon!
CODE
The Mythical Business Layer
CODE
WTF.js?!?
CODE
Box Up Your Microservices with F#!

CODE TRAINING

We have a long history in training. We have done training events of various sizes, ranging from individual training and mentoring, all the way up to small conferences. More recently, we have focused on just a few kinds of specific training events:

1. Classroom style training (online and on-site) of our own products, such as CODE Framework. This is usually free of charge.
2. Classroom style training for standard technologies such as WPF, ASP.NET, or VFP Conversion. Normally this is multi-day classes that the customer pays for.
3. Lecture style events at various locations (often at Microsoft offices) that are normally half-day. Our State of .NET Events are the prime example of this. These events are free.
4. "Day of" lectures. These are day-long training sessions with beefy content held at various locations, often at Microsoft offices. Examples of this are "A Day of ASP.NET", "A Day of Cloud/Azure", "A Day of Angular", and so on. These events are usually free of charge.

As you can see, many of our training events are free of charge. While in the past, training used to be a revenue stream for us, it has now turned into mostly a marketing effort. It is an important aspect of our business.

CODE STAFFING

CODE Staffing is our strategic staffing and staff augmentation effort. It is an important part of our business that we expect to grow quite a bit in the future. In fact, you may be reading this book for the very reason of being part of the CODE Staffing effort.

The main difference between our approach and what "conventional" staffing companies do is that we use the power of the CODE ohana to back up our staffing efforts. Customers do not just get a "warm body", but they work with someone who can always reach out to our overall network to get questions answered and problems solved. Thus our customers not only get access to the consultant whom they work directly with, but also to our entire network with all our contacts.

CODE FRAMEWORK

CODE Framework is our framework product that helps developers more efficiently build advanced business applications. The framework is perhaps the most well known for its features for Windows Desktop applications, but the web and services are other key components as well.

CODE Framework is open source and free of charge. However, CODE Framework is still very important for our business, since some of the CODE Framework users become consulting customers.

All CODE ohana developers should be familiar with CODE framework. For more information, check out **www.codemag.com/framework**. Also, make sure you check out our training materials and hands-on labs, as well as live and recorded training sessions.

VFP CONVERSION

Legacy conversion projects, and especially conversions from former Visual FoxPro projects, have been, and will continue to be, an important part of our business. Most people are surprised by how many legacy projects are still running in production and in dire need of being replaced.

Relationship with Other Companies

We have a long history of grooming relationships with other companies, such as Microsoft. We actively pursue these relationships and go out of our way to meet with people from other companies and invite them to lunch or dinner or support them in their own goals in various ways. For example, Microsoft evangelists have internal goals to be involved in community events, and they are always receptive in us helping them with this.

This is something our entire ohana needs to be aware of. Let us know if you need help in engaging with your local representatives of other companies. Also let us know if you have new ideas about other companies we should approach and be involved with.

Technologies & Services

This is the section where we would love to have a list of all our services we offer and technologies we use. This is where we should talk about our long -time relationship with Microsoft, and how we have been doing .NET for ages and our background in HTML that goes back to the very early days. The trouble is technologies are constantly changing, and thus so are our services and offerings. For this reason, it is best to go to our web site (**www.codemag.com**) and look at the full list of services and technologies there.

Some of our Clients

eog resources
VMS
commonGOALS software
Winpark THE EXACT RIGHT WAY
RISC
emsi
CDPHE CO
CAMERON
SOLOVUE
BRIDGESTONE
GILLIG
Salisbury CLOUD-BASED MEDICAL BILLING & PM SOFTWARE SERVICE
EMERSON
MACK SOFTWARE SOLUTIONS, INC.
gamma IT SOLUTIONS
ups
PARCEL PRO A UPS Capital Company
ROYAL Your Perishable Products Specialist
SA Software Answers

Working Remotely

In the past few years, we have transitioned to being mostly a virtual company. Yes, we still have a brick and mortar office in Houston, and we will continue to do so.

However, for many team members, telecommuting is the standard approach. We support this, and will work together with you to establish a setup that works well for all of us. However, there are certain expectations we have from you. We need you to have real a place to work where you can attend meetings, record videos, do phone calls with clients, etc and it is up to you to set this up. We need you to have good internet connectivity.

Beyond that, things are relatively straightforward. We for employees expect you to be online during working hours and use our standard communication tools (such as GoToMeeting, Skype, MS Teams, and email).

For our employees we offer great flexibility with our working hours, however note that flexibility does not mean part-time! We expect you to put in a regular workweek. We also expect you to adjust your working hours so you can be present for meetings. Be sure your manager knows what you would like your standard working hours to be. Setting such expectations avoids a lot of misunderstandings and hardship.

If you are a contractor, this section doesn't apply to you. You are an independent vendor and you are expected to set your own hours and have your own location of business. Of course, as you work with us on projects, it is simply good business sense to be available on standard communication channels, and we most certainly expect you to show up for scheduled meetings. Having a good working relationship will make projects more fun and productive and will also cause more business to flow your way.

Teambuilding

We primarily use GoToMeeting, Skype, MS Teams, and email to keep in touch with each other remotely. This works pretty well day-to-day, but we feel it is extremely important to periodically bring the whole team together face to face. Our goal for this is twice a year.

We invest a lot of money into our team-building events. After all, a group of remote people working together do not just accidentally end up being a strong team. It takes active effort, which is why we find these events to be important and worth spending time and money on. For our employees, we strongly recommend that you participate. For contractors, it is of course up to you, since you are an independent vendor. However, you are invited, and it certainly helps to be as connected to the team as possible, when it comes to picking up more business from us.

Our team-building events include:

The Company Cruise – We take the "Party Bus" from our offices to Galveston where we go on a 4 day, EPS paid company cruise. Hilarity ensues. :-)

Dev Summit and Christmas Party – 2 days of both technology and company meetings that are both fun and informative. Event wraps up with our company Christmas Party – often at Fogo de Chao! (Don't even both trying to compete with Otto!)

Going to Hawaii

You are probably thinking "Let's cut to the chase already! When do I get to go to Hawai'i?"

Well, that is a good question. Many of EPS staff members have come to Hawai'i to work on various projects, but truth be told, it is not a very common thing. If you want to go to Hawai'i on vacation, we can help you figure it out and get you some special deals. We even often discuss how we could have future company get-togethers in Maui, but so far, we haven't figured out a way that is financially feasible and works for everyone. But who knows?

For now, we always have our company cruise to Mexico ;-)

CODE Terminology

Sometimes we speak our own language! Here are some terms you may hear around the office:

Austria – Land of schnitzel, the Sound of Music, and Mozart. Oh, and this is where Markus is from and where EPS started. In fact, we still have an Austrian office.

ClockWork – Our internal timekeeping system. All employees and contractors are required to track their time in ClockWork since this is where we pull the data from to bill our clients.

CODE Consulting – Our main business involves consulting and writing custom software.

CODE Focus – Special issues of CODE Magazine, focusing on various technologies. These are often produced together with the creator of the technology (such as Microsoft).

CODE Framework – Our internal development framework, which we also make available to outsiders, free of charge and open-source.

CODE Magazine – Our awesome developer magazine.

CODE Staffing – Our strategic staff augmentation arm, which differs drastically from other staffing companies due to our extensive networks of professionals, backed by DevNet.

CODE Training – Our training arm. Now mostly a marketing initiative. In the past, we did this as a profit center and put on anything from small individual training events all the way up to small conferences.

Conversion Projects – Many of our software projects involve rewriting outdated software. Often the existing software was originally written in Visual FoxPro.

DevNet – CODE's Developer Network. A network of contractors who we have an established relationship with and who we can contact when we need resources on a project.

DEP – Developer Education Program. Periodic free developer training sessions that all employees and contractors are invited to attend. These sessions are given by various developers and are recorded so developers can watch them later.

Ellen Whitney – Co-owner of EPS, and in charge of all marketing and finances (among other things). You do not want to piss off the person in charge of finances.

Maui – The Hawaiian island Markus, Ellen (and Rick) live on.

Lunch with CODE – One of our marketing efforts where we go to companies and do a free lecture during lunch. They pick the topic and the food. **www.codemag.com/lunch**

Markus Egger – Markus is the founder of EPS and other ventures.
Mike Yeager – CEO of EPS in Houston. Mike runs things. You don't want to piss off the person who runs things.

Milos Solution Platform – The predecessor of CODE Framework.

Rick Strahl – Technically not part of EPS (he owns West Wind Technologies), but has been Markus' "partner in crime" for years. Rick and Markus started *CODE Magazine* together. Rick is also the master of all things web. Or, as Markus likes to tell him: "all he does all day is send strings back and forth over the internet".

State of .NET – One of our marketing efforts where we do a free, half-day lecture about future technologies and which ones developers should be learning. We often do these lectures at Microsoft's offices. **www.codemag.com/stateofdotnet**

Tower48 – A separate company owned by Markus Egger, Ellen Whitney, Mike Yeager, Simon Ferguson and Pierce Adkins that provides digital escrow services. – **www.tower48.com**

Visual FoxPro – Developer tool popular in the past. Visual FoxPro is/was owned by Microsoft (after they acquired Fox Software) and was used to squash Borland. Remember Borland? Well... if you don't, that is due to FoxPro. We have a long history with FoxPro. Markus even worked on the FoxPro team for a while. We still get business from converting old FoxPro applications to more modern technologies.

Wikinome (formerly GeneSys) – Startup project funded by EPS Software that deals with processing genomic sequencing data, led by Otto Dobretsberger. **www.wikinome.com**

Xiine – A digital reading platform we championed before digital reading platforms were a thing. Unfortunately, a customer ended up owing us a lot of money, so we didn't have the funds to push this the way we had planned. **www.xiine.com**

Unicorns – Sorry, we can't tell you **all** our secrets just yet!!