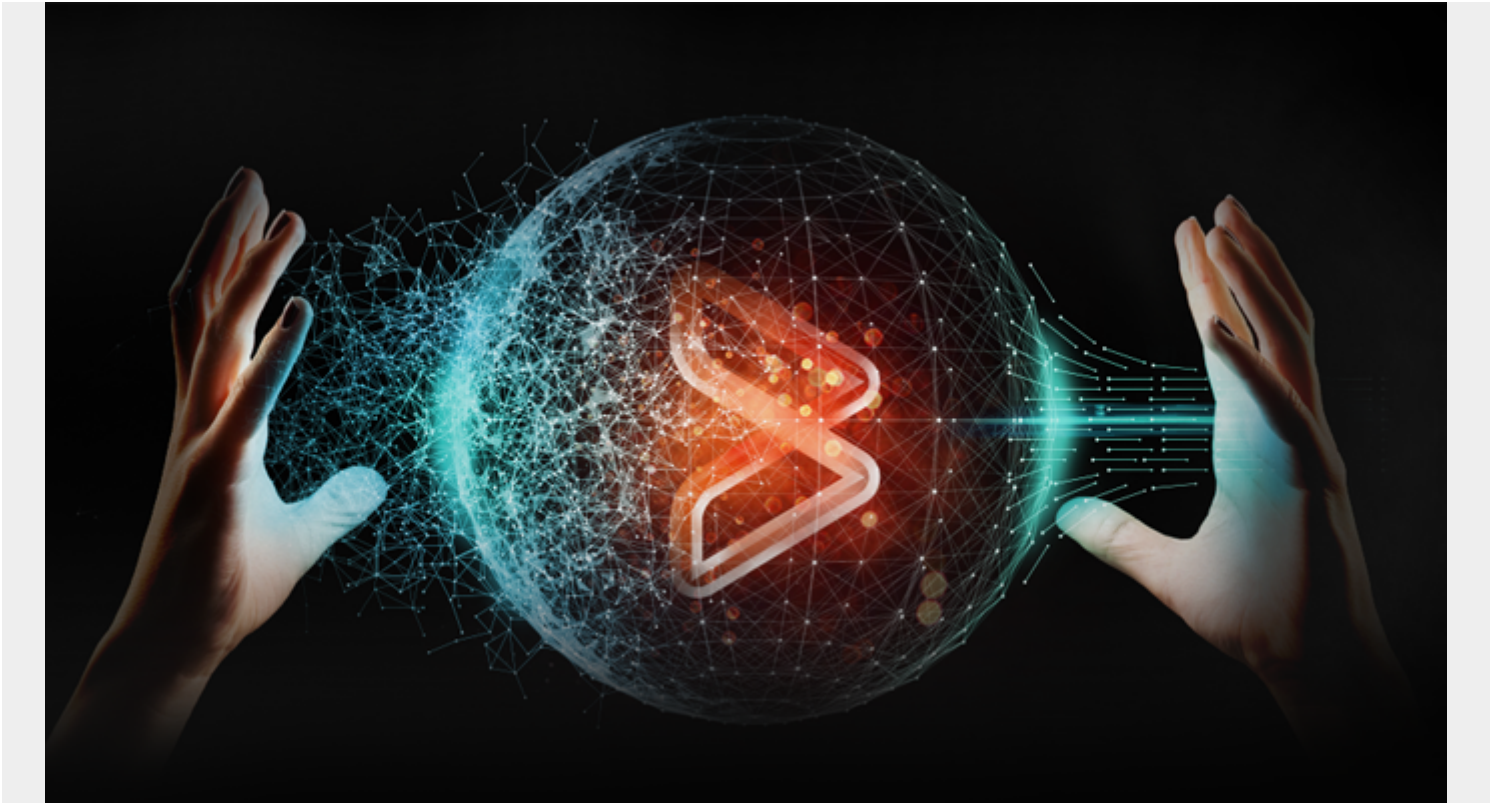# MAKING ROOM FOR CREATIVITY IN MAINFRAME SOFTWARE DEVELOPMENT

Overview: Many mainframe software development teams fear creativity because it clashes with their rigid processes. But it must be embraced for digital success and can be balanced well with Agile Development.

There is an unnecessary fear that if we think of mainframe software development as being creative, it will lead to chaos and missed deadlines. I can tell you this won't be the case.

A few months back while visiting customers in Europe, I found some downtime to attend the London Podcast Festival. I'm an avid podcast listener, particularly interested in audio dramas like _Wooden Overcoats_ (my favorite), _We Fix Space Junk_ and _Wolf 359_. I enjoy the creativity that produces a quirky English village, a spaceship or space station. The festival was an exciting opportunity to speak with and learn from actors, writers, producers, directors, etc. from these shows.

But I'm also a software guy, a Product Owner for Compuware ISPW, BMC's source code management solution. So, while I got to learn a few new things from the podcasting creatives I admire, you might wonder what that has to do with building software.

Actually, I was struck by the parallels between how people build software and how people build

audio drama podcasts. Here's a little insight into how podcasters work, based on my conversations with them:

- Podcast productions involve a lot of people and coordination—there are scripts, recording facilities, sound effects, distribution, etc.
- They have set schedules for releasing episodes, often weekly or every other week.
- They need to meet these deadlines and consistently provide a good experience.
- Their audiences expect consistent quality, or else they leave.

Podcasters provide room for creativity in a structured way with the knowledge that, ultimately, they're creating something for other people—their audience is their customers.

Here's what [David K. Barnes](#), the creator of *Wooden Overcoats*, told me:

"For writers, the old joke about deadlines is that a project will always expand to fit the time you have to do it in. A script might require a week to write, but if you have three weeks available to you, then you'll find a reason why it takes three weeks!

"Podcasting, however, requires a certain level of discipline to ensure that content is created on time for your audience. That discipline is even more important if your podcast is released throughout the year."

The key is to balance creativity with your other requirements. The best structure available to provide this balance in mainframe software development is Agile Development, which allows you to produce quality software fast and efficiently on time.
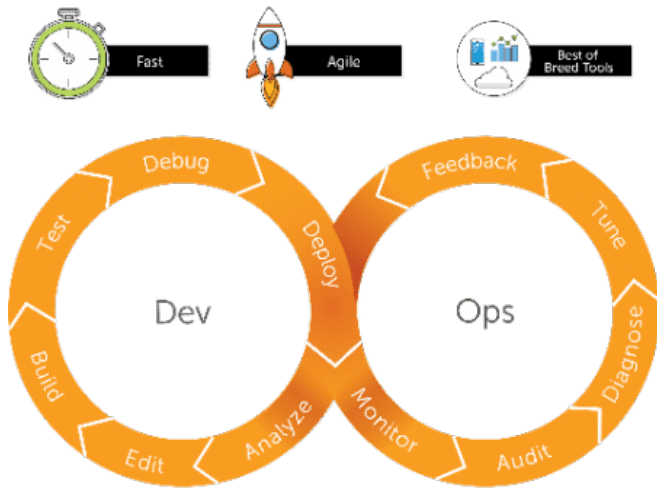
Being Agile to Be Creative

Many in mainframe software development fear everyone will accomplish less work if they're spending time being creative. But the very deadlines David and his crew stick to can actually create pressure that releases creativity.

Similarly, using two-week sprints—quick, two-week development cycles in Agile Development—you deliver smaller pieces of innovation to customers at a faster cadence and quickly respond to their feedback. Two-week sprints also help you sharpen your creativity, as you're constantly researching, designing, prototyping, coding, testing, demoing and getting feedback—you're constantly learning and improving.

Contrast this with a structure like Waterfall, where these activities are spread apart, allowing developers to get rusty and the value of the work their doing to depreciate as it sits, waiting to be delivered 18 months in the future.

OPEN SYSTEMS

Fast | Agile | Best of Breed Tools

Debug · Feedback · Test · Tune · Deploy · Build · Diagnose · Edit · Analyze · Monitor · Audit

Dev — **Ops**

Continuous Delivery

MAINFRAME SYSTEMS

Slow | Waterfall | Monolithic Tools

Requirements → Design → Develop → Test → Analysis → Deploy

12 - 18 Month Cycle

Imagine spending months, maybe even years, creating a podcast drama of 30 episodes that you release all at once, after every episode has been written, recorded and edited to near perfection. You'll likely have fewer people listening to the entire series. And who's to say your drama will be as relevant or as popular as it could have been were you releasing episodes once every two weeks, taking listener feedback and using it to tweak the story of the drama as you wrote it?

Podcasters are agile! At least the ones I listen to are. And this agility allows them to remain creative while meeting other requirements that aren't so creative but are entirely necessary.

For mainframe software development, Agile has a strict structure which needs to be adhered to because of the tight deadlines. It also provides new opportunities, like with the sprint retrospective meeting where teams take an honest look at the previous sprint. This is where you look for ways to improve, no matter how small. You test them in the next sprint and then once again reflect and see how they worked. This constant reflection and search for improvement is an example of how Agile Development helps you foster creativity.

How to Foster Creativity in Your Teams

Bottom line, creativity is called for in mainframe software development. Here are some ways to encourage it:

- Highlight creative solutions to difficult problems. Have the team member explain how they came up with the idea.
- Encourage exploration, such as learning a new language or technique.
- Call out when creativity is needed. There is no clear answer here—let's get creative!
- Bring others (non-engineers from across the organization) into your sprint reviews. Their ideas can help inspire out-of-the-box solutions.
- Perhaps most important let people know there won't be a punishment for taking creative risks.

Just like you can't ignore customer needs and deadlines, you can't ignore creativity if your organization wants to produce the best solutions. If you can't do that, meeting your other requirements will begin to matter less, because your organization will begin to matter less as more creative competitors begin to matter more to your customers.
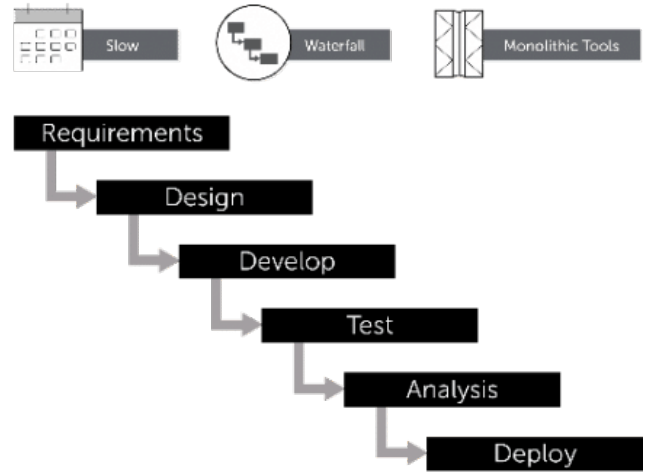
So, don't fear creativity in mainframe software development; embrace it.