

Cool Tools for PHP Development

Ben Ramsey

Essex Web Dev • 21 July 2021

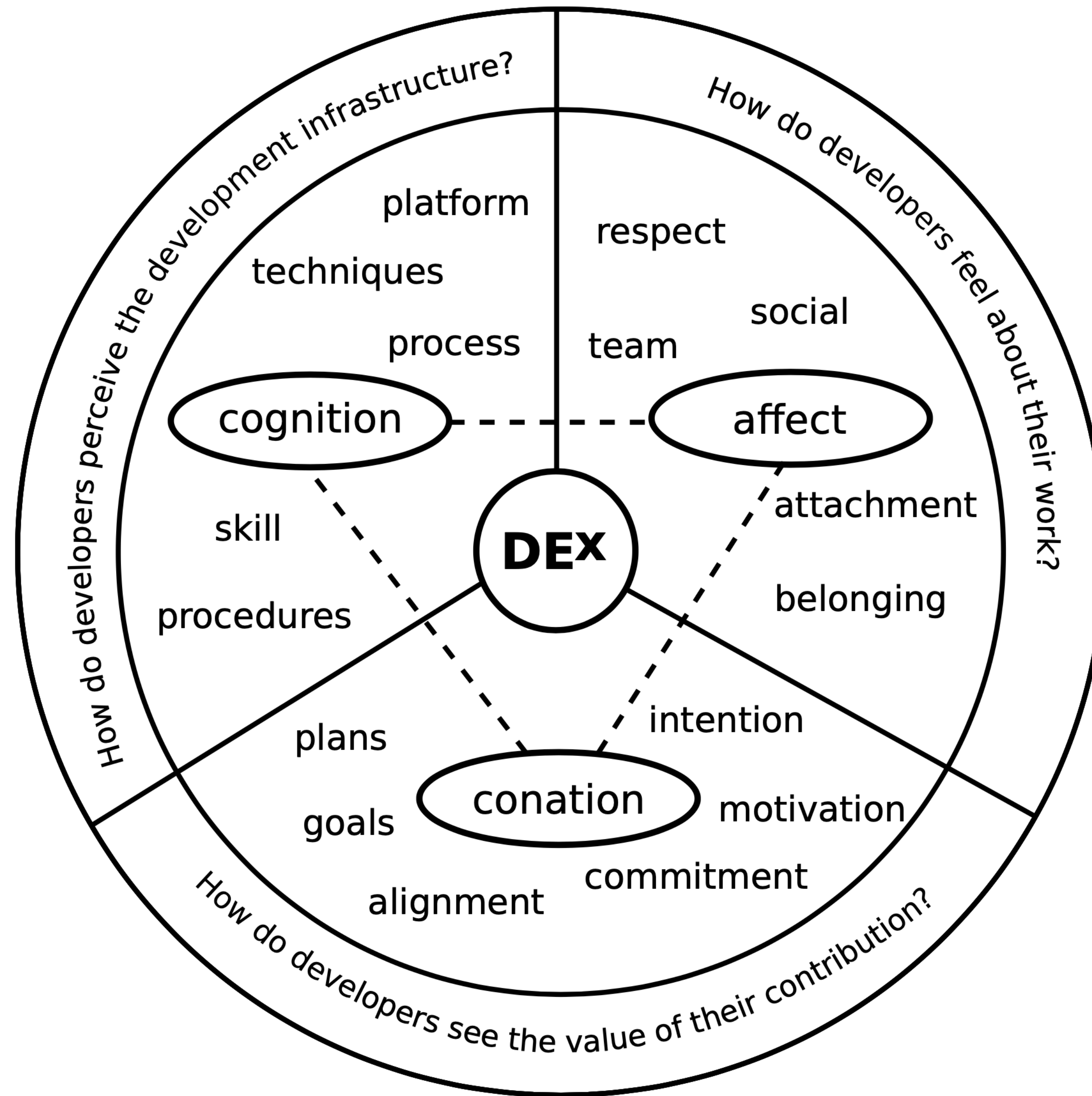


What is DX?

DEVELOPER EXPERIENCE



“DE^x consists of experiences relating to all kinds of artifacts and activities that a developer may encounter as part of their involvement in software development. These could roughly be divided into experiences regarding i) development infrastructure (e.g. development and management tools, programming languages, libraries, platforms, frameworks, processes, and methods), ii) feelings about work (e.g. respect, attachment, belonging), and iii) the value of one’s own contribution (e.g. alignment of one’s own goals with those of the project, plans, intentions, and commitment).”



Developer experience

CONCEPTUAL FRAMEWORK

Cognition

How do developers perceive the development infrastructure?

Conation

How do developers see the value of their contribution?

Affect

How do developers feel about their work?

How do you perceive the infrastructure?

Do your tools get in the way?

Are they a joy to use?

The better experience a developer has with their tools, the more efficient and happy they'll be.



Cool tools



Cool tools

WHAT ARE THEY?

The tools you need to do your job

The tools that make your job easier

The tools that bring joy to your work



Cool tools

OUR FOCUS

How can you create better workflows for the tools you already use?

Programming is a “team sport.” How can you share these workflows with your team?

In other words, how can you improve your DX?



Live coding!



Recap

WE COVERED...

Using Composer scripts to stitch together workflows and share with team/contributors

Using Composer plugins to combine workflows into commands that can be shared across projects

Using CaptainHook to ensure team/contributors follow workflows seamlessly



Learn more

Composer scripts:

getcomposer.org/doc/articles/scripts.md

Composer plugins:

getcomposer.org/doc/articles/plugins.md

CaptainHook:

captainhookphp.github.io/captainhook



**Lots of
cool tools!**



atoum

Behat

bitexpert/captainhook-infection

CaptainHook

Code Scrawl

Codeception

composer outdated -D

composer validate

composer-lock-diff

composer-normalize

composer-unused

ComposerRequireChecker

Danger PHP

dePHPend

Deptrac

doctrine orm:validate-schema

Dredd

Exakat

GrumPHP

Infection

laminas/automatic-releases

Lando

Latte syntax checker

madewithlove/license-checker

[mamazu/documentation-validator](#)

[Markdown Link Linter](#)

[mlocati/docker-php-extension-installer](#)

[Neon checker](#)

[Parse: A PHP Security Scanner](#)

[Pest](#)

[Phan](#)

[Phinx: Simple PHP Database Migrations](#)

[PHIVE](#)

[php -l](#)

[PHP Architecture Tester](#)

[PHP Copy/Paste Detector](#)

[PHP CS Fixer](#)

[PHP Extensions Finder](#)

[PHP Insights](#)

[PHP Magic Number Detector](#)

[PHP Mess Detector](#)

[PHP Parallel Lint](#)

[PHP Quality Assurance](#)

[PHP Security Advisories Database](#)

[PHP VarDump Check](#)

[PHPBench](#)

[PHPCompatibility](#)

[PHPCSExtra](#)

[phpDocumentor](#)

[phpDox](#)

[PHPStan](#)

[Phpsu: Synchronisation Utility](#)

[PHPUnit](#)

[PHP_CodeSniffer](#)

[Prettier](#)

[Psalm](#)

[ramsey/composer-install](#)

[ramsey/composer-repl](#)

[ramsey/conventional-commits](#)

[ramsey/devtools](#)

[Rector](#)

[Roave Backward Compatibility Check](#)

[Roave Security Advisories](#)

[roave/no-floaters](#)

[roave/no-leaks](#)

[roave/you-are-using-it-wrong](#)

[Robo](#)

[Slevomat Coding Standard](#)

[symfony check:security](#)

[symplify/easy-coding-standard](#)

[UpToDocs](#)



**And many more at
packagist.org**

Thanks!

 [@ramsey](https://twitter.com/@ramsey)

 @ramsey@phpc.social

 github.com/ramsey

 dev.to/ramsey

 ben@benramsey.com

© 2021 Ben Ramsey

This work is licensed under a
[Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

All photos are from Unsplash and used according to the [Unsplash License](https://unsplash.com/terms).

