Microsoft Word Author Guidelines for EUROGRAPHICS

Proceedings Manuscripts

D.W. Fellner1,2 and S. Behnke2

1TU Darmstadt & Fraunhofer IGD, Germany  
2Institut für ComputerGraphik & Wissensvisualisierung, TU Graz, Austria

# Abstract

The ABSTRACT is to be in fully-justified italicized text, between two horizontal lines, in one-column format, below the author and affiliation information. Use the word “Abstract” as the title, in 9-point Times, boldface type, left-aligned to the text, initially capitalized. The abstract is to be in 9-point, single-spaced type. The abstract may be up to 3 inches (7.62 cm) long.

Leave one blank line after the abstract, then add the subject categories according to the ACM Classification Index (see http://www.acm.org/class/1998/).

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and Curve Generation

# Introduktion

# Vad är DevOps?

Enligt “A Survey of DevOps Concepts and Challenges” skriver dem “DevOpsis [sic] a collaborative and multidisciplinary organizational effort to automate continuous delivery of new software updates while guaranteeing their correctness and reliability.” [1] DevOps är så mycket mer än att jobba på ett visst sätt med ett visst antal verktyg. ”Devops is a way of thinking and a way of working. It is a framework for sharing stories and developing empathy, enabling people and teams to practice their crafts in effective and lasting ways. It is part of the cultural weave that shapes how we work and why. Many people think about devops as specific tools like Chef or Docker, but tools alone are not devops. What makes tools “devops” is the manner of their use, not fundamental characteristics of the tools themselves.” DevOps är alltså mer än bara ett sätt, en metod att producera mjukvara efter. ”Devops is not just another software development methodology. Although it is related to and even influenced by software development methodologies like Agile or XP, and its practices can include software development methods, or features like infrastructure automation and continuous delivery, it is much more than just the sum of these parts. While these concepts are related and may be frequently seen in devops environments, focusing solely on them misses the bigger picture—the cultural and interpersonal aspects that give devops its power.” [2] Låt oss därför börja med den historiska och kulturella delen av devops innan vi går in på den tekniska och praktiska delen.

## DevOps kultur och historia

Datorer reflekterar människors gärningar genom vad vi programmerar datorer till att göra. Vi använder datorer för att underlätta vår vardag. På samma sätt har devops kommit till. Det handlar om att när man jobbar med något, så lär man sig mer om produkten och arbetsmiljön, och detta leder till att vilja effektivisera sin arbetsprocedur så mycket som möjligt.

Författarna bakom *The DevOps Handbook* ger en utmärkt förkaring: ”DevOps and its resulting technical, architectural, and cultural practices represent a convergence of many philosophical and management movements. While many organizations have developed these principles independently, understanding that DevOps resulted from a broad stroke of movements, a phenomenon described by John Willis (one of the co-authors of this book) as the “convergence of DevOps,” shows an amazing progression of thinking and improbable connections. There are decades of lessons learned from manufacturing, high reliability organization, high-trust management models, and others that have brought us to the DevOps practices we know today.”

DevOps bygger på olika metoder ifrån olika organisationer ifrån olika tider: ”DevOps relies on bodies of knowledge from Lean, Theory of Constraints, the Toyota Production System, resilience engineering, learning organizations, safety culture, human factors, and many others. Other valuable contexts that DevOps draws from include high-trust management cultures, servant leadership, and organizational change management.” [3]

## DevOps praktiskt och tekniskt

För att arbeta efter devops (d**ev**elopment och **op**eration**s**)[4] så handlar det inte bara om att skapa mjukvara, utan också hur de anställda går tillväga för att utveckla mjukvaran. ”The heart of devops starts with people working not only as groups but as teams with a desire for mutual understanding. This can be described as a compact that teams will work together, communicate their intentions and the issues that they run into, and dynamically adjust in order to work toward their shared organizational goals.” [5]

Konceptet devops utgår ifrån att leverera en tjänst eller en produkt till en kund. ”In DevOps, we typically define our technology value stream as the process required to convert a business hypothesis into a technology-enabled service that delivers value to the customer.” När man börjar jobba på kundens idé, jobbar utvecklingsteamet efter en typisk agil eller iterativ process och börjar implementera detta i kod. Koden tillhandahålls genom versionshantering. Varje ändring i koden integreras och testas med resten av mjukvarusystemet. Integreringen innebär att bygga applikationen i ett byggsystem som bygger om applikationen för att se att alla komponenter samarbetar, och testerna, inkluderat säkerhetstester, testar att applikationen ska göra vad den är menat att göra. Som nämnt i förra citeringen, värde (”value”) sker endast när tjänsten körs i produktion. När man skickar applikationen till en produktionsserver, så behöver man försäkra sig om att driftsättningen inte orsakar fel och störningar såsom service-, driftavbrott eller säkerhetsfel, bland annat[6].

# Hur arbetar man inom DevOps?

Effektiv devops innefattar flera steg, från utveckling och test till produktion. Man brukar följa en procedur när man utvecklar mjukvara, även känt på engelska som ”software development methodology”. Inom denna kategori brukar det agila (en. “Agile”) arbetssättet efterföljas. Det betyder dock inte att man alltid måste följa just detta arbetssätt. Det handlar mer om miljön koden arbetar inom. ”Devops is not so rigidly defined as to prohibit any particular methodology. While devops arose from practitioners who were advocating for Agile system administration and cooperation between development and operations teams, the details of its practice are unique per environment.”

När man skriver kod vill man hantera detta på ett effektivt sätt. Man vill kunna snabbt ladda upp koden och testa den; man vill kunna gå tillbaka till en tidigare version ifall koden innehåller buggar; teamet vill arbeta på olika delar av koden i applikationen. Detta kallas för versionshantering. Ett välkänt versionshanteringssystem som många utvecklare använder heter Git.

När man skriver koden vill man även testa den[7]. Man brukar ha ett automatiserande testramverk som går igenom olika testfall man skrivit för koden för att se att de nya funktioner man skapat går igenom. Ett sådant testramverk kan tex. vara PHPUnit för kod skriven i PHP. Säkerhetstester brukar också inkluderas, för att kolla sårbarheter i applikationen eller koden. Sådana säkerhetstester kan bestå av Bandit för Python. Denna sortens säkerhetstest brukar definieras som ”Static Application Security Testing (SAST), eller bara Static Code Analysis”. ”Dynamic Application Security Testing (DAST) letar efter sårbarheter i webbapplikationer genom att skanna och utföra attacker på applikationen.” [8] För det här ändamålet kan man använda ZAP.

När man känner att man är klar tex. med en viss funktion som gått igenom de lokala testerna, vill man gärna se hur implementeringen i applikationen beter sig i sin helhet som om den skulle köras på produktionsservern. Man brukar då pusha koden till sitt versionshanteringssystem. Efter man pushat koden, brukar en viss sida, som tex. TravisCI, bygga applikationen. Detta kallas för Continuous integration (CI). En sådan sida/tjänst känner av varje push man begått och bygger applikationen och kör testerna. Allt detta sker automatiskt. När inga fel uppstår förbereder man applikationen som ska köras på live servern för Continuous delivery (CD).

I CD stadiet befinner sig applikationen redo för att publiceras på produktionsservern för att nyttjas av slutanvändaren.

## Type-style and fonts

Wherever Times is specified, Times Roman may also be used. If neither is available on your word processor, please use the font closest in appearance to Times that you have access to. Only Type-1 fonts will be accepted.

MAIN TITLE. The title should be in Times 17-point, boldface type and centered. Capitalize the first letter of nouns, pronouns, verbs, adjectives, and adverbs; do not capitalize articles, coordinate conjunctions, or prepositions (unless the title begins with such a word). Leave two blank lines after the title.

AUTHOR NAME(s) and AFFILIATION(s) are to be centered beneath the title and printed in Times 9-point, non-boldface type. This information is to be followed by two blank lines.

The ABSTRACT is to be in a one-column format. The MAIN TEXT is to be in a two-column format.

MAIN TEXT. Type main text in 9-point Times, single-spaced. Do *not* use double-spacing. All paragraphs should be indented 1 em (the length of the dash in the actual font). Make sure your text is fully justified – that is, flush left and flush right. Please do not place any additional blank lines between paragraphs. Figure and table captions should be 9-point Times boldface type as in Figure 1.

Long captions should be set as in Figure 1 or Figure 3.

sampleFig-gray

Figure 1: *This figure serve as an example of long caption requiring more than one line. It is not typed centered but aligned on both sides.*

Figures which need the full textwidth can be typeset as Figure 3.

Callouts should be 9-point Times, non-boldface type. Initially capitalize only the first word of section titles and first-, second-, and third-order headings.

FIRST-ORDER HEADINGS. (For example, **1. Introduction**) should be Times 9-point boldface, initially capitalized, flush left, with one blank line before, and one blank line after.

SECOND-ORDER HEADINGS. (For example, **2.1. Language**) should be Times 9-point boldface, initially capitalized, flush left, with one blank line before, and one after. If you require a third-order heading (we discourage it), use 9-point Times, boldface, initially capitalized, flush left, preceded by one blank line, followed by a period and your text on the same line.

The headline (authors / title) must be shortened if it uses the full two column width of the main text. There must be enough space for the page numbers. Please use “et al.” if there are more than three authors and specify a shortened version for your title.

## Footnotes

Please do *not* use footnotes at all!

## References

List all bibliographical references in 9-point Times, single-spaced, at the end of your paper in alphabetical order. When referenced in the text, enclose the citation index in square brackets, for example [Lou90]. Where appropriate, include the name(s) of editors of referenced books.

For your references please use the following algorithm:

* **one** author: first 3 chars plus year – e.g.[Lou90]
* **two**, **three** or **four** authors: first char of each family name plus year – e.g.[FH93] or [KSS97] or [LFTG97]
* **more than 3** authors: first char of family name from first 3 authors followed by a ‘\*’ followed by the year –e.g.[BFH\*98] or [FvDF\*93]

For BibTeX users a style file eg-alpha.bst is available which uses the above algorithm.

## Illustrations, graphs, and photographs

All graphics should be centered.

sampleFig

For all figures please keep in mind that you **must not** use images with transparent background!

Figure 2: *Here is a sample figure.*

If your paper includes images, it is very important that they are of sufficient resolution to be faithfully reproduced.

To determine the optimum size (width and height) of an image, measure the image’s size as it appears in your document (in millimeters), and then multiply those two values by 12. The resulting values are the optimum *x* and *y* resolution, in pixels, of the image. Image quality will suffer if these guidelines are not followed.

Example 1: An image measures 50 mm by 75 mm when placed in a document. This image should have a resolution of no less than 600 pixels by 900 pixels in order to be reproduced faithfully.

Example 2: Capturing a screenshot of your entire 1024×768 pixel display monitor may be useful in illustrating a concept from your research. In order to be reproduced faithfully, that 1024×768 image should be no larger than 85 mm by 64 mm (approximately) when placed in your document.

## Color

**Please observe:** as of 2003 publications in the proceedings of the Eurographics Conference can use color images throughout the paper. No separate color tables are necessary.

However, workshop proceedings might have different agreements! Figure 3 is an example for creating color plates.

## Embedding of Hyperlinks / Typesetting of URLs

*(This is not applicable if using Microsoft Word.)*

## PDF Generation

Your final paper should be delivered as a PDF document with all typefaces embedded.

*(LaTeX-related details omitted.)*

We recommend to download and install the version of the “CMW” Adobe Acrobat Distiller job options file appropriate for your operating system and version of Acrobat from the following URL:

http://www.cadmusmediaworks.com/index2.html

in the “(Operating System)/Applications/Distiller Settings” folder. The “CMW” job options file embeds all typefaces and does not downsample or subsample images when creating the PDF document.

## Copyright forms

You must include your signed Eurographics copyright release form when you submit your finished paper.We MUST have this form before your paper can be published in the proceedings.

## Conclusions

Please direct any questions to the production editor in charge of these proceedings.

# References

[1] The ACM Digital Library, *A Survey of DevOps Concepts and Challenges*, (https://dl.acm.org/doi/10.1145/3359981), hämtad 2020.

[2] Effective DevOps, *Chapter 1. The Big Picture*, (https://learning.oreilly.com/library/view/effective-devops/9781491926291/ch01.html), hämtad 2020.

[3] The DevOps Handbook, *Part I, A BRIEF HISTORY*, (https://learning.oreilly.com/library/view/the-devops-handbook/9781457191381/DOHB-pt\_01\_text.xhtml), hämtad 2020.

[4] TechnologyAdvice, *DevOps - development and operations*, (https://www.webopedia.com/TERM/D/devops\_development\_operations.html), hämtad 2020.

[5] Effective DevOps, *Chapter 2. What Is Devops?*, (https://learning.oreilly.com/library/view/effective-devops/9781491926291/ch02.html), hämtad 2020.

[6] The DevOps Handbook, *1 Agile, Continuous Delivery, and the Three Ways*, (https://learning.oreilly.com/library/view/the-devops-handbook/9781457191381/DOHB-ch\_01.xhtml), hämtad 2020.

[7] Effective DevOps, *Chapter 4. Foundational Terminology and Concepts*, (https://learning.oreilly.com/library/view/effective-devops/9781491926291/ch04.html), hämtad 2020.

[8] dbwebb, *SAST vs. DAST*, (https://dbwebb.se/kurser/devops-v1/kmom05#sast-dast), hämtad 2020.

[] sida/bok titel, titel/kapitel, (länk), hämtad 2020.

[] sida/bok titel, titel/kapitel, (länk), hämtad 2020.

[] sida/bok titel, titel/kapitel, (länk), hämtad 2020.