A PLASTIC SCM CASE STUDY

# HOW KO_OP USES PLASTIC SCM

## TO ACCELERATE PRODUCTION AND REDUCE CHAOS

Unity's version control solution unites artists and engineers with better asset and source code management.

KO_OP

→ **THE CHALLENGE**
Optimizing asset management and version control with engaged artists.

→ **THE PLATFORMS**
Android, iOS, PC, PlayStation

→ **PROJECT STAFF**
24

→ **COMPANY**
Montreal, Canada

# KO_OP: A PLASTIC SCM CASE STUDY

How does a worker-owned cooperative studio get both artists and engineers aligned on a production process? As a version control platform, Git initially seemed adequate for programmers, but artists found it challenging and frustrating, and this led KO_OP's version control policies to slip. Mistakenly duplicated, overwritten, and mismanaged assets slowed production and frustrated everyone.

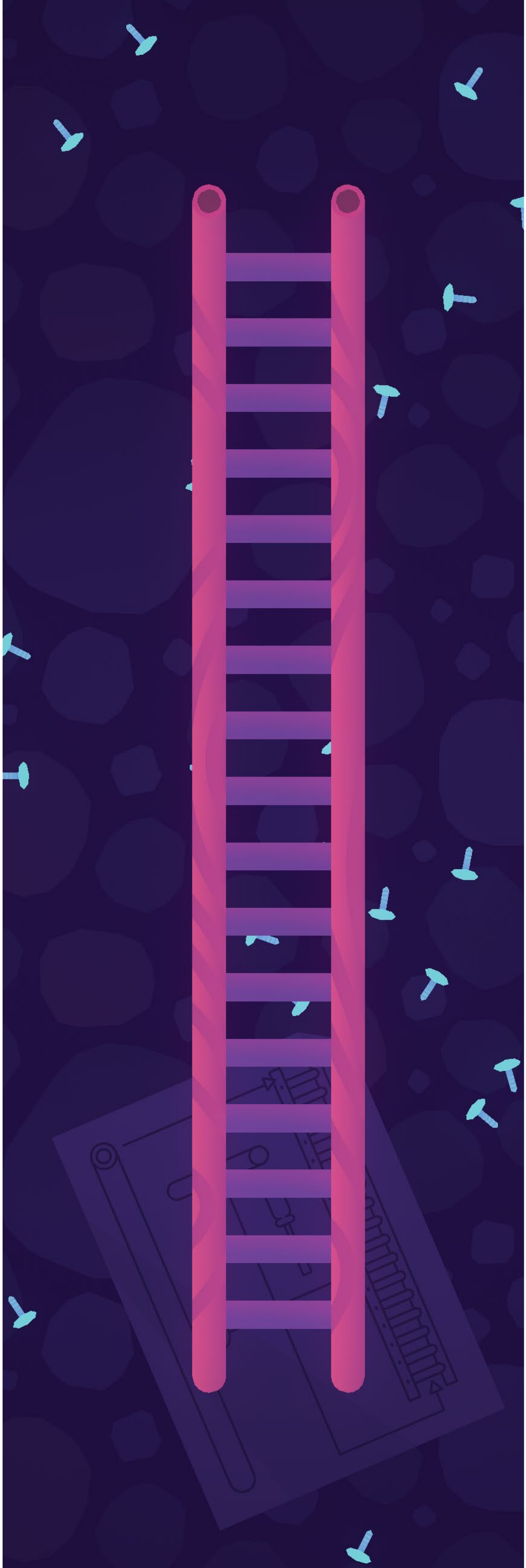
# ONE VERSION CONTROL PLATFORM BOTH ARTISTS AND ENGINEERS SUPPORT

Artists and engineers work independently, and both are essential to creating a successful game. Tracking every art asset and every line of code – from creation through revisions and release – is as critical for efficient development workflows as counting money is for a bank. Yet having a version control platform that only some contributors use isn't nearly enough, which is why KO_OP turned to Unity Plastic SCM Cloud Edition for version control across the studio.

© 2021 Unity Technologies

# THE RESULTS

- Onboarded the entire team to a single version control system, moving from a handful of programmers and two integrators on Git to everyone using Plastic SCM
- Reduced the number of tech support calls dramatically – from twice a week for Git to a couple of times per month – eliminating a major pain point
- Used Plastic more widely and harnessed more advanced and complex features because of reduced cost and risk



# COOPERATIVE BY DESIGN

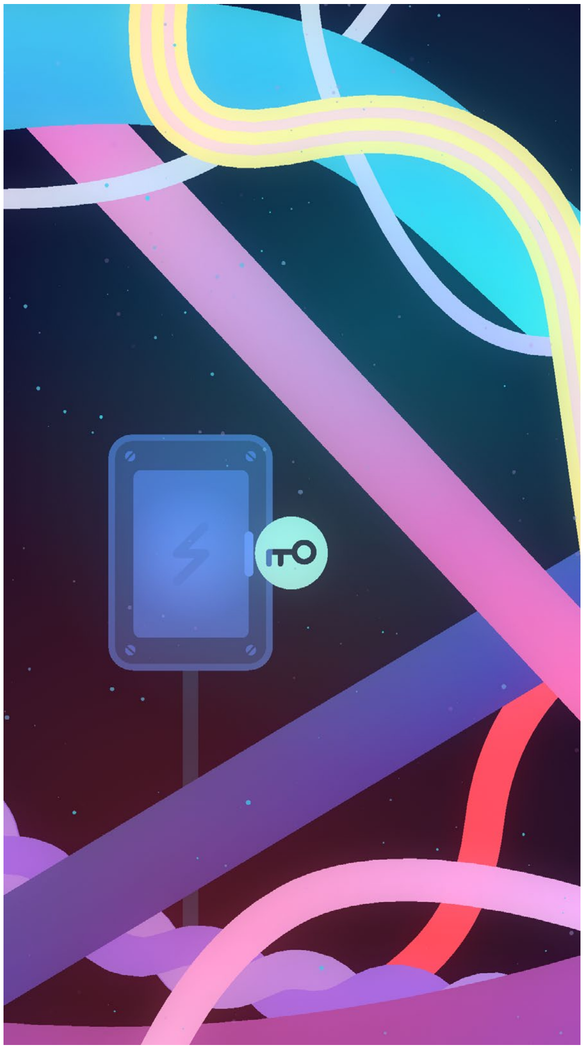
KO_OP is an artist-run studio where all full-time workers are equal owners – each member receives the same pay, and they make design, development, and business decisions as a team. Since its founding in 2012 by studio director Saleem Dabbous and programmer Bronson Zgeb, they've been experimenting with games and interactive art using Unity. Production highlights so far include *Laura Croft GO* expansion *Mirror of Spirits*, AR/VR-enabled *GNOG*, and their most recent release, the Apple Arcade game *Winding Worlds*. They're currently preparing to launch *Goodbye Volcano High*, a highly anticipated narrative adventure game.

"This studio, this company, exists to support the people who are part of it, not the other way around," as Dabbous explained in a recent *Vice* profile. "KO_OP is there for us to take advantage of whatever resources it awards us – be it healthcare or this opportunity to create a certain piece of art that we want to see out there in the world."

*Goodbye Volcano High*

© 2021 Unity Technologies

GNOG



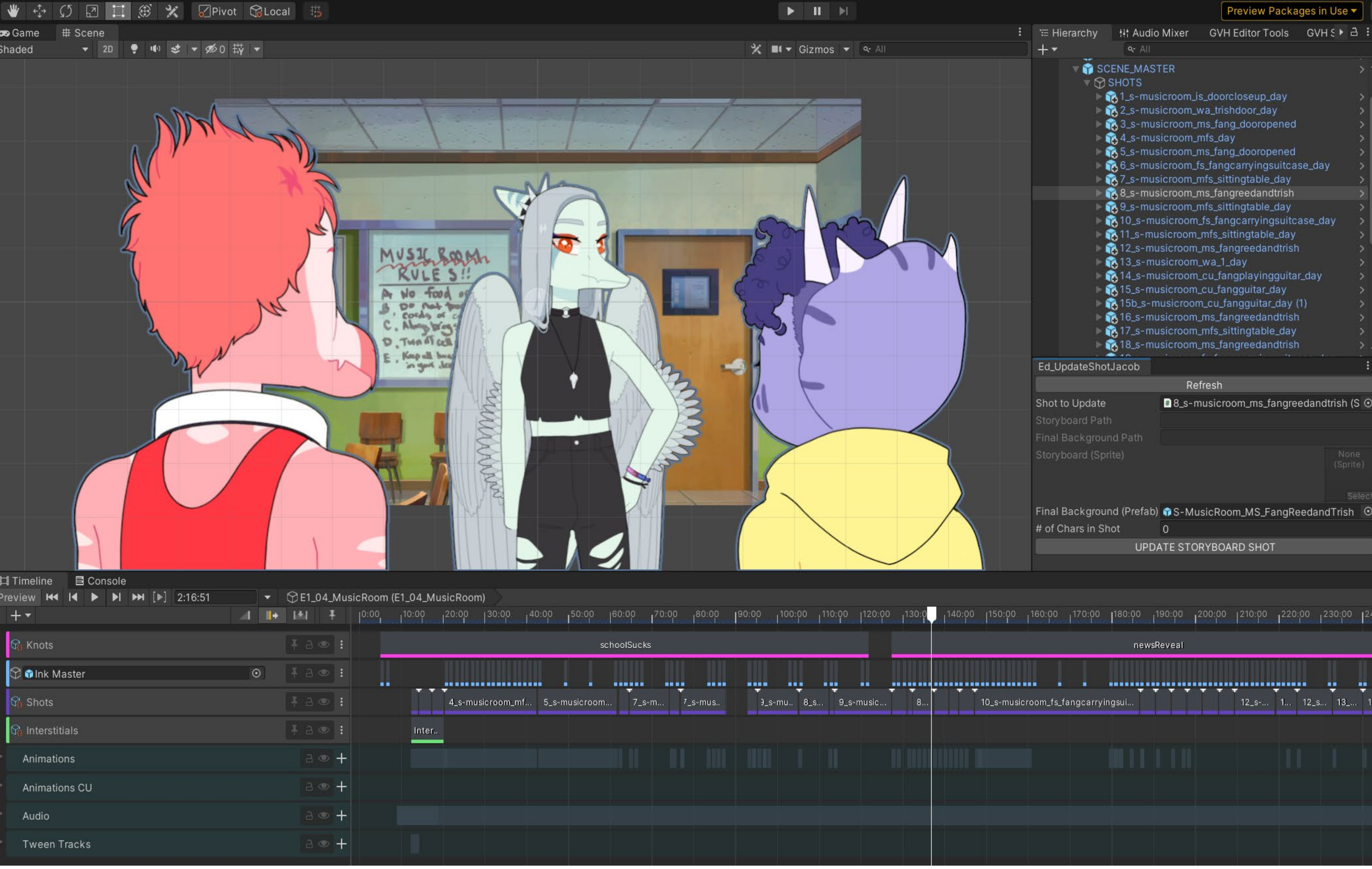
# GETTING BY WITH GIT

Artists, designers, and engineers have very different tasks in game development, yet their efforts are wholly intertwined. Visibility about what's needed when, what's been done, what's changed, what's in progress, and the like is critical to working together smoothly.

KO_OP initially used Git for version control, but the team found that they were missing this panoramic visibility. Workflows were chaotic, and there were constantly questions about who was working on what, or whether it was OK to tweak a particular scene. Artists would unknowingly work on the same file independently, forcing engineers to deal with merge conflicts.

Programmers had relied on Git for source code management most of their careers, but for less-technical artists, Git was largely a mystery written in another language. One engineer was tasked with helping other team members with their version control issues, but it was a constant uphill battle. And when the pandemic dispersed everyone to remote work, communication just got worse. And asset errors proliferated.

© 2021 Unity Technologies


Pivot
Local
Preview Packages in Use
Game
Scene
Shaded
2D
Gizmos
Hierarchy
Audio Mixer
GVH Editor Tools
SCENE_MASTER
SHOTS
1_s-musicroom_is_doorcloseup_day
2_s-musicroom_wa_trishdoor_day
3_s-musicroom_ms_fang_dooropened
4_s-musicroom_mfs_day
5_s-musicroom_ms_fang_dooropened
6_s-musicroom_fs_fangcarryingsuitcase_day
7_s-musicroom_mfs_sittingtable_day
8_s-musicroom_ms_fangreedandtrish
9_s-musicroom_mfs_sittingtable_day
10_s-musicroom_fs_fangcarryingsuitcase_day
11_s-musicroom_mfs_sittingtable_day
12_s-musicroom_ms_fangreedandtrish
13_s-musicroom_wa_1_day
14_s-musicroom_cu_fangplayingguitar_day
15_s-musicroom_cu_fangguitar_day
15b_s-musicroom_cu_fangguitar_day (1)
16_s-musicroom_ms_fangreedandtrish
17_s-musicroom_mfs_sittingtable_day
18_s-musicroom_ms_fangreedandtrish
Ed_UpdateShotJacob
Refresh
Storyboard Path
Final Background Path
Storyboard (Sprite)
None (Sprite)
UPDATE STORYBOARD SHOT
Timeline
Console
Preview
Knots
Ink Master
Shots
Interstitials
Animations
Animations CU
Audio
Tween Tracks
schoolSucks
newsReveal
Inter..
MUSIC ROOM RULES!!

Shot to Update 8_s-musicroom_ms_fangreedandtrish (S

Final Background (Prefab) S-MusicRoom_MS_FangReedandTrish

# of Chars in Shot 0

2:16:51 E1_04_MusicRoom (E1_04_MusicRoom)

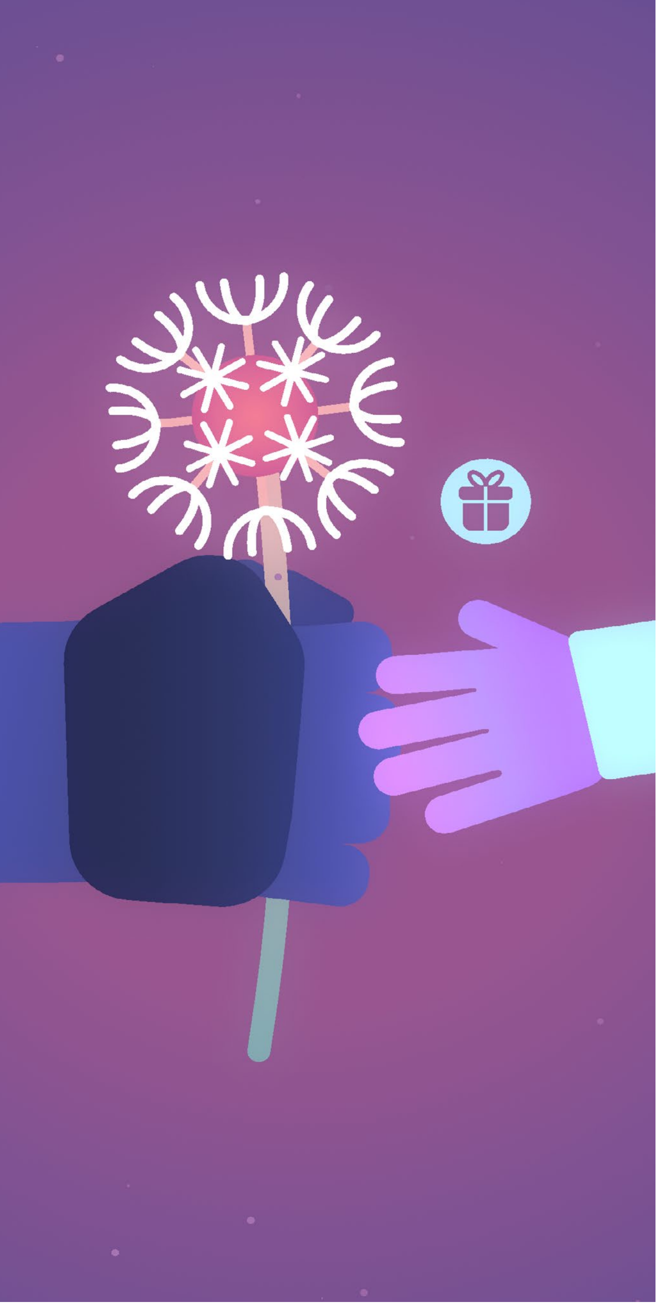
# TIME FOR A CHANGE

KO_OP was in the middle of their *Goodbye Volcano High* project. They were encountering obstacles with their version control, and it was clear they needed a fix. As a Unity shop, trying Plastic SCM was an obvious first step. It's an easy-to-use, platform-agnostic version control system that lets programmers, artists, and designers collaborate more efficiently. According to Dabbous, "the Plastic visual interface for locking assets made their availability clear and could eliminate disruptions." They ran some tests involving other specific pain points and soon concluded that Plastic was the solution.

Plastic provides two tools to help teams migrate from Git, one for a partial switchover and one for a full migration. KO_OP was willing to go all-in, and they adjusted to the new UI slowly but surely. The studio very much appreciated Plastic's documentation, which outlined best practices and methods for efficiency.

"It wasn't 'this tool does this, this tool does that, now figure it out,'" says Dabbous. "Plastic recommended and showed how to set up a branch model at a much more granular and effective level than what we were used to."

© 2021 Unity Technologies



# GLUON GUI EMPOWERS VISUAL THINKERS

The team's engineers were fine with Git, perhaps because they're accustomed to making sense of code-filled screens. On the other hand, artists prefer approachable, visually rich tools where color and shapes communicate what's going on. Plastic accommodates both teams with Gluon, a full artist-oriented GUI and workflow that lets artists pick up files easily and handle large binaries. According to designer Jacob Blommestein, "Plastic Gluon was a lovely surprise. The artists didn't have to know anything about branching or merging. They just added their .psd files, and the versioning was transparent."

The engineers loved Plastic's branching visualization. Dabbous says, "Plastic is easy to parse and much easier to navigate than Git." The GUI for code reviews built into the Plastic SCM client surprised the engineers. But when they understood that they had it, they realized that they very much needed it. It was much easier to check out different parts and branches of a project than before. "In Git, it felt like if someone touched the wrong thing, everything would break," continues Dabbous. "In Plastic, people can jump around the project in ways that won't be destructive."

> "PLASTIC GLUON WAS A DELIGHTFUL SURPRISE. THE ARTISTS DIDN'T HAVE TO KNOW ANYTHING ABOUT BRANCHING OR MERGING. THEY JUST ADDED THEIR .PSD FILES, AND THE VERSIONING WAS TRANSPARENT.
>
> — Jacob Blommestein, Designer, KO_OP

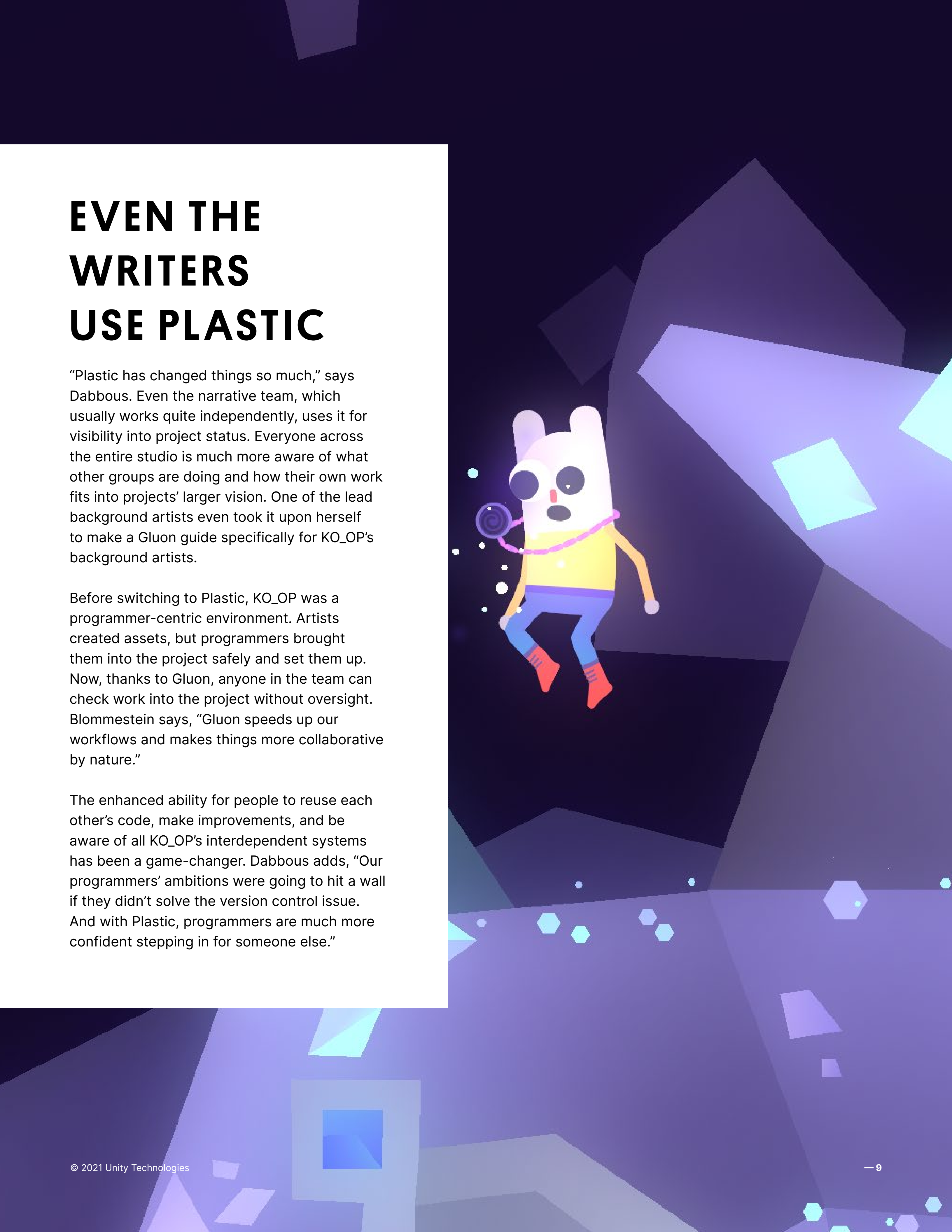
© 2021 Unity Technologies

# EVEN THE WRITERS USE PLASTIC

“Plastic has changed things so much,” says Dabbous. Even the narrative team, which usually works quite independently, uses it for visibility into project status. Everyone across the entire studio is much more aware of what other groups are doing and how their own work fits into projects’ larger vision. One of the lead background artists even took it upon herself to make a Gluon guide specifically for KO_OP’s background artists.

Before switching to Plastic, KO_OP was a programmer-centric environment. Artists created assets, but programmers brought them into the project safely and set them up. Now, thanks to Gluon, anyone in the team can check work into the project without oversight. Blommestein says, “Gluon speeds up our workflows and makes things more collaborative by nature.”

The enhanced ability for people to reuse each other’s code, make improvements, and be aware of all KO_OP’s interdependent systems has been a game-changer. Dabbous adds, “Our programmers’ ambitions were going to hit a wall if they didn’t solve the version control issue. And with Plastic, programmers are much more confident stepping in for someone else.”



© 2021 Unity Technologies

# EXPANDING COLLABORATION ACROSS THE TECH STACK

*Goodbye Volcano High* will be KO_OP's biggest game ever, which is why they were willing to risk a mid-project switch in their version control system to ensure the game's timely launch. The risk paid off – the entire company is now aligned on a unified workflow, and they're working less as independent contributors and more as a team.

Implementing Plastic, a significantly different approach to version control than Git, gave KO_OP an opportunity to reboot and redefine their entire production line. This meant not just tracking versions and assets better but reconfiguring how projects went from conception through implementation, testing, and integration.

It also meant finding ways to integrate Plastic and their other essential communication tools, Slack and Jira, as well as build automation. Compulsively creative as they are, the KO_OP programmers have started work on a series of DevOps tools to connect the three apps with capabilities such as automatic change notifications. Dabbous says, "After our success with Plastic, we felt we should take the next step and improve collaboration across the board."

© 2021 Unity Technologies

Unity®