

Aktualizacja rozwoju: Zbliżamy się do wydania wersji Release Candidate 3.0

<https://www.gimp.org/news/2024/10/05/development-update/>

2024-10-05

przez [zespół GIMP](#)

To krótka aktualizacja rozwoju na temat naszych postępów w kierunku pierwszego release candidate dla **GIMP 3.0**. Niedawno osiągnęliśmy kamień [milowy zamrożenia ciągu](#). Oznacza to, że nie będzie już żadnych zmian w tekście widocznym dla użytkownika (takich jak etykiety i komunikaty GUI), aby tłumacze mogli pracować nad ostatecznymi tłumaczeniami dla wersji 3.0. Postęp w rozwiązywaniu problemów blokujących dla wersji 3.0 spowolnił w okresie letnim, ponieważ konserwator i niektórzy z głównych deweloperów zachorowali po [konferencji Libre Graphics Meeting](#). Dlatego jesteśmy „spóźnieni” w porównaniu z pierwotnie szacowanymi terminami. Na szczęście wszyscy czują się teraz znacznie lepiej, a prace zostały wznowione na dobre! W chwili pisania tego tekstu osiągnęliśmy 96% ukończenia kamienia [milowego 3.0 RC1](#), a pozostało 11 problemów.

Będziemy mieli o wiele więcej do przekazania w poście z wiadomościami o GIMP 3.0 RC1. Jednak oto kilka najważniejszych rzeczy, które pokażą, nad czym pracowaliśmy przez ostatnie kilka miesięcy:

- [API](#)
 - [Tworzenie interfejsu graficznego wtyczki](#)
 - [Aktualizacje Script-Fu](#)
 - [Opcje eksportu](#)
- [Inwazja przestrzeni barw](#)
- [Aktualizacje edycji nieniszczącej](#)
- [Biblioteki rodzinne GIMP: ctx, babl i GEGL](#)
- [Ulepszenia procesu kompilacji](#)
- [Integracja darktable](#)
- [Dokumentacja](#)
- [Naprawiono błędy](#)
- [GSoC2024](#)
- [Zespół projektowy](#)
- [Co dalej?](#)

API

Finalizowanie API jest kluczowym zadaniem dla GIMP 3.0. Ponieważ dodawaliśmy nowe funkcje i ulepszaliśmy istniejące podczas rozwoju, musieliśmy dokonać „łamliwych” zmian w publicznym API. Oznacza to, że jeśli zewnętrzny programista przeniósł swoją wtyczkę lub skrypt 2.10 do używania w API GIMP 2.99.16, może to nie działać z API 2.99.18 z powodu dalszych zmian. [Jednak po wydaniu wersji 3.0 każda funkcja, która znajduje się w publicznym API, musi nadal działać we wszystkich przyszłych wydaniach GIMP 3.](#) Musimy więc zrobić to dobrze przed pierwszym wydaniem 3.0! Większość zmian API jest niewidoczna dla osób niebędących programistami, więc nie będziemy ich tutaj szczegółowo opisywać. Chcielibyśmy jednak podzielić się kilkoma, aby zilustrować trwające prace:

Tworzenie interfejsu graficznego wtyczki

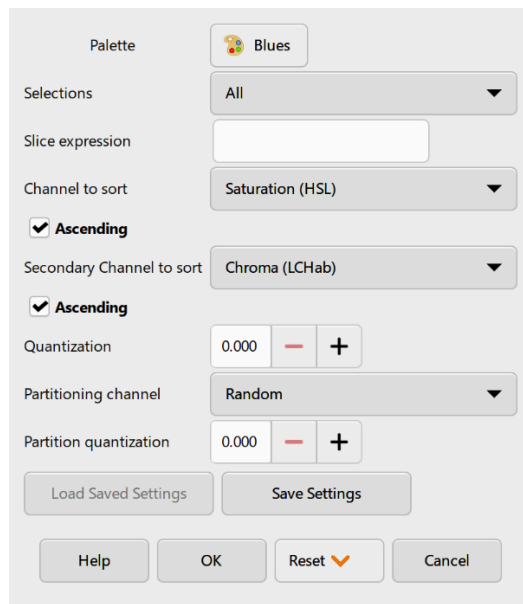
W kilku poprzednich wersjach nasze wewnętrzne wtyczki zostały przeniesione do nowego `GimpProcedure` i `GimpProcedureDialog` API. Ta aktualizacja automatycznie zapisuje ostatnio używane ustawienia, pozwalając użytkownikom na ich zresetowanie lub przywrócenie „domyślnych ustawień fabrycznych”, kiedy tylko chcą. API `GimpProcedureDialog` pozwala również deweloperom na automatyczne tworzenie GUI na podstawie zdefiniowanych przez nich ustawień.

Do niedawna jednak ta funkcja tworzenia GUI była w pełni dostępna tylko dla wtyczek C – inne języki wtyczek, takie jak Python, nie mogły generować niektórych widżetów, takich jak pola rozwijane i przyciski radiowe.

Jednak od wydania 2.99.18 mogliśmy udostępnić pełne API wszystkim obsługiwanym językom wtyczek. Deweloperzy wtyczek Python mogą zobaczyć więcej przykładów korzystania z tego nowego API w naszym repozytorium sekcji wtyczek Python:

<https://gitlab.gnome.org/GNOME/gimp/-/tree/master/plugin-ins/python>

Gdy API zostanie w pełni ustabilizowane, zaktualizujemy naszą witrynę dla deweloperów <https://developer.gimp.org/> o więcej samouczków dotyczących korzystania z tego i innych API w Twoich wtyczkach.



Przykład wygenerowanego okna dialogowego (wtyczka Python do sortowania palet)

Aktualizacje Script-Fu

Lloyd Konneker zorganizował i wdrożył wiele udoskonaleń w naszej bibliotece kodu Script-fu. Dla programistów skryptów, `script-fu-register` została wycofana i zastąpiona dwiema nowymi funkcjami: `script-fu-register-procedure` dla skryptów ogólnych i `script-fu-register-filter` dla skryptów przetwarzania obrazu.

Te dwie nowe funkcje skryptów również używają `GimpProcedureDialog` API, więc programiści skryptów będą mieli dostęp do tego samego automatycznego tworzenia GUI, o którym wspomniano w poprzedniej sekcji. Możesz przejrzeć nasz przewodnik w toku:

<https://gitlab.gnome.org/GNOME/gimp/-/blob/master/devel-docs/GIMP3-plugin-porting-guide/script-fu-author-guide.md> aby dowiedzieć się, jak możesz korzystać z tych nowych funkcji podczas przenoszenia skryptów wtyczek 2.10.

Opcje eksportu

Długotrwałym żądaniem funkcji było eksportowanie obrazów z różnymi ustawieniami, przy jednoczesnym pozostawieniu oryginalnego obrazu bez zmian. Na przykład umożliwienie użytkownikom eksportowania obrazu w kilku różnych rozmiarach.

Nowa klasa `GimpExportOptions` stanowi podstawę do wdrożenia tego w przyszłych wersjach 3.x.

Uprościliśmy sposób eksportowania obrazów za pomocą interfejsu API wtyczki i przenieśliśmy większość kodu ustawień eksportu do parametru `GimpExportOptions`. Ta zmiana pozwoli nam dodawać nowe typy ustawień eksportu i funkcje po wersji 3.0 bez konieczności wprowadzania zmian w ich własnym kodzie przez twórców wtyczek. Jako miły efekt uboczny, ta praca naprawiła również pewne istniejące niespójności między eksportowaniem obrazu z interfejsu graficznego GIMP a eksportowaniem ze skryptu.

Inwazja przestrzeni barw

Drugim pozostałym obszarem pracy nad wersją 3.0 jest dokończenie projektu inwazji przestrzeni kolorów. Naszym celem jest, aby informacje o przestrzeni kolorów i profilu kolorów były powiązane z pikselami we wszystkich aspektach programu GIMP, od płótna do interfejsu graficznego i wszędzie pomiędzy. Jest to ważne dla artystów, aby zachować spójność kolorów na wszystkich urządzeniach i monitorach, których używają. Pierwszą połowę tej pracy ukończył **Jehan w wydaniu 2.99.18**. Od tego czasu naprawialiśmy nieuniknione błędy wynikające z tak dużej zmiany, jednocześnie dostosowując resztę programu GIMP do przestrzeni kolorów. Ta praca pokrywa się ze zmianami w interfejsie API, ponieważ kilka naszych funkcji kodu nadal zakładało, że kolory znajdują się w przestrzeni kolorów sRGB.

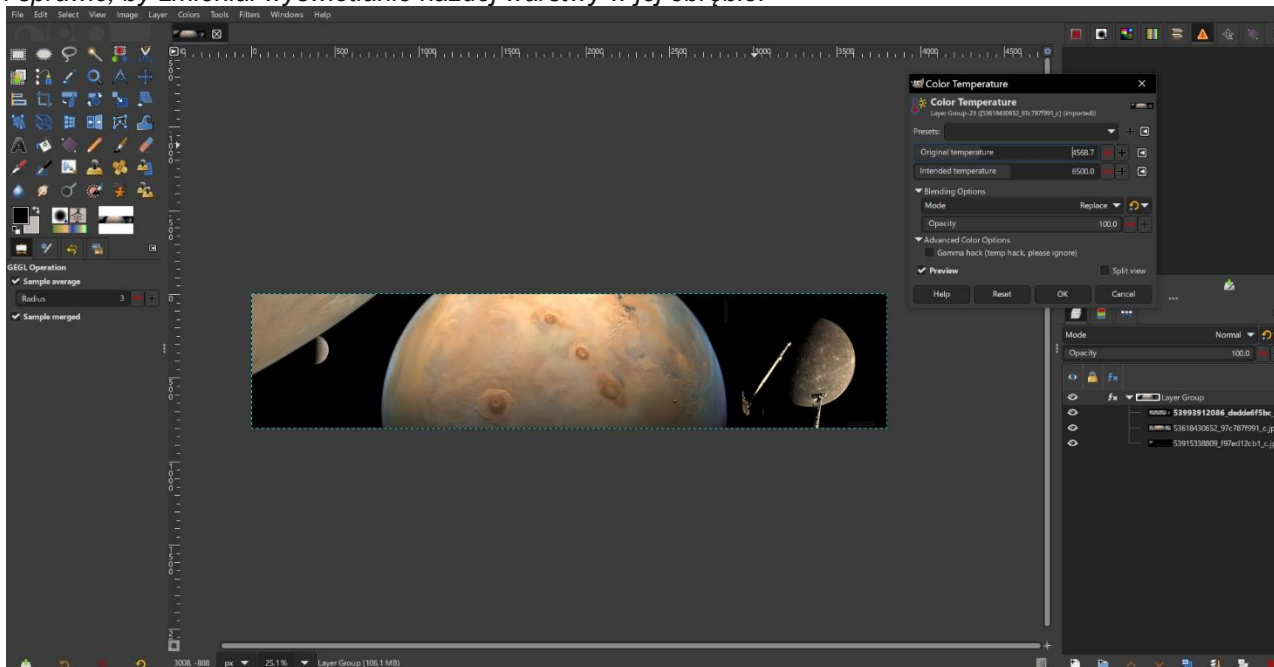
Ponadto dokonaliśmy przeglądu istniejącego algorytmu kolorów, aby upewnić się, że jest poprawny i działa wydajnie. **Øyvind Kolås** i **Elle Stone** zapewнили nam świetny wgląd i pomoc w tym procesie. Chcemy się upewnić, że pliki projektu GIMP 2.10 XCF wyglądają tak samo po otwarciu w GIMP 3.0, ale chcemy również skonfigurować infrastrukturę, aby poprawić dokładność trybów warstw i innych aspektów GIMP w przyszłości.

Aktualizacje edycji nieniszczącej

Od czasu wprowadzenia nieniszczących filtrów w GIMP 2.99.18

<https://www.gimp.org/news/2024/02/21/gimp-2-99-18-released/#initial-non-destructive-layer-effects>

otrzymaliśmy wiele świetnych opinii i raportów o błędach od wczesnych użytkowników. Na podstawie tych raportów naprawiliśmy wiele błędów związanych z kopiowaniem, wklejaniem i aktualizowaniem filtrów, a także poprawiliśmy ogólną stabilność kodu efektów. Tymczasowa ikona filtra została również zastąpiona bardziej intuicyjnym **f_x** projektem nowego współpracownika **Denisa Rangelov** (stworzonego za pomocą programu do grafiki wektorowej [Inkscape](#), innego projektu FLOSS, który gorąco polecamy). Oprócz bieżących poprawek błędów, wdrożyliśmy również **filtry nieniszczące w grupach warstw**. Teraz możesz dodać filtr regulacyjny, taki jak Brightness-Contrast (lub dowolny inny efekt warstwy) do grupy i sprawić, by zmieniła wyświetlanie każdej warstwy w jej obrębie.



Przykład filtra edycji nieniszczącej temperatury barwowej stosowanego do grupy warstw. Zdjęcia autorstwa [Andrea Luck](#), Uznanie autorstwa (CC BY 2.0)

Biblioteki rodzinne GIMP : ctx, babl i GEGL

Øyvind również ciężko pracował nad [ctx](#) przez ostatnie kilka miesięcy, w tym nad poprawą przenośności dla różnych platform (na wszystkich typach architektur, libc i OS), poprawą wydajności i masowym profilowaniem i testowaniem rozmytym projektu. Dla przypomnienia, ctx jest jednym z najnowszych ambitnych projektów w rodzinie GIMP, do renderowania wektorów 2D i serializacji. Choć nie jest jeszcze konieczne często używany w samym GIMP, może ugotować drogę do przyszłych prac nad większą liczbą możliwości wektorowych w naszym oprogramowaniu.

Oczywiście, wszystko to dzieje się przy jednoczesnym utrzymaniu [babl](#) i [GEGL](#), naszego silnika konwersji kolorów i opartego na grafach frameworka przetwarzania pikseli. Te dwie biblioteki nie otrzymują ostatnio znaczących zmian, pomimo całej pracy wykonanej z Color Space Invasion i projektami edycji nieniszczącej, co jest całkiem dobrym znakiem stabilnego oprogramowania w dobrej kondycji!

Ulepszenia procesu kompilacji

Bruno Lopes ciężko pracował, aby ulepszyć nasze procesy kompilacji na wszystkich platformach. Jego ciągła praca pomogła zredukować redundancję i nieefektywność w naszym procesie rozwoju, instalatorach systemu Windows i dystrybucjach Flakpak. Przygotowuje również nową wersję naszego instalatora Microsoft Store, która będzie lepiej zintegrowana z platformą, a jak wspomniano w poprzednim poście informacyjnym <https://www.gimp.org/news/2024/05/28/experiments-appimage/>, eksperymentuje z wersją AppImage programu GIMP. Możesz również podziękować Bruno za jego pracę nad aktualizacją dokumentacji kompilacji na naszej [stronie internetowej dla deweloperów](#).

Integracja z darktable

Chociaż GIMP nie przetwarza natywnie obrazów RAW, mamy wtyczki, które umożliwiają wysyłanie i pobieranie obrazów za pomocą świetnego oprogramowania do przetwarzania zdjęć RAW FLOSS, takiego jak [darktable](#) i [RawTherapee](#). Na początku tego roku darktable zaktualizowało swoje publiczne API, którego GIMP używa do skonfigurowania tego połączenia, co spowodowało, że wtyczka przestała działać. Na szczęście **Hanno Schwalm** i inni programiści darktable [współpracowali z nami](#), aby stworzyć specyficzne dla GIMP API, które powinno być bardziej stabilne w przyszłości. Naprawdę doceniamy współpracę z programistami darktable w celu przywrócenia tego połączenia!

(Należy pamiętać, że zaktualizowany interfejs API nie jest jeszcze dostępny w programach GIMP 2.10.38 i GIMP 2.99.18). Na razie można używać [darktable w wersji 4.6](#) i starszych z programem GIMP jako obejścia problemu)

Dokumentacja

Ze wszystkimi nowymi zmianami i ulepszeniami w programie GIMP 3.0 podręcznik pomocy wymaga wielu aktualizacji w stosunku do wersji 2.10. **Jacob Boerema** objął kierownictwo nad tym projektem, aby zaktualizować zrzuty ekranu i tekst, a także dodać nowe sekcje.

To obszar, w którym **każdy może pomóc** bez konieczności pisania ani jednej linijki kodu!

Możesz przejrzeć nadchodzącą dokumentację na naszej stronie testowej podręcznika pomocy:

<https://testing.docs.gimp.org/>

Jeśli zauważysz, że czegoś brakuje lub jest to nieaktualne, możesz o tym napisać w [naszym systemie śledzenia problemów](#). Jeśli chcesz pomóc dalej, możesz również samodzielnie rozwiązać problem i przesłać prośbę o scalenie.



Home | About | Help us | Releases | Tutorials | Donate

Search docs.gimp.org

GIMP Documentation

GIMP 2.99 Online Manuals

Podgląd podręcznika pomocy programu GIMP 3.0; ilustracja: [Aryeom](#) (CC by-sa 4.0)

Naprawiono błędy

Od czasu kamienia milowego zamrożenia funkcji skupiliśmy się na naprawianiu jak największej liczby błędów przed pierwszym wydaniem. Obejmują one wszystko, od starszych błędów, które istniały w GIMP 2.10, po te ostatnio utworzone podczas implementacji wszystkich nowych funkcji dla GIMP 3.0. Szczególne podziękowania dla **Andersa Jonssona**, **Andre Klappera**, **Lloyda Konnekera** i **Sama Lestera** za ich rozległą pracę nad wyszukiwaniem i naprawianiem tych błędów!

Wcześni użytkownicy i testerzy również dostarczyli cenne raporty o błędach, więc jeśli natknąłeś się na błąd w wersjach rozwojowych, zgłoś go w naszym systemie śledzenia problemów:

<https://gitlab.gnome.org/GNOME/gimp/-/issues> .

GSoC2024

Po raz kolejny wzięliśmy tego lata udział w GSoC. Tym razem mieliśmy szczęście pracować z trzema studentami-współpracownikami. Ze względu na okoliczności wymienione powyżej ich projekty zostały nieco ograniczone w porównaniu z naszymi początkowymi planami. Mimo to wszyscy trzej studenci wykonali świetną pracę!

- **Idriss Fekir** kontynuował pracę nad udoskonaleniem narzędzia tekstowego z GSoC 2023. Jego praca pokrywała się również z udoskonalaniem przestrzeni kolorów, np. rozwiązywaniem problemów z kolorem tekstu, ponieważ uwzględniał on przestrzeń kolorów.
- **Cheesequake** przeprowadził wstępne badania i projekt, aby ostatecznie przenieść nasz GtkTreeView GUI do GTK4. Pomagał również w [naprawie wielu błędów](#) w naszym nieniszczącym kodzie edycyjnym.
- **Varun Samaga BL** pracował nad ulepszeniem kodu OpenCL w GEGL. OpenCL przyspiesza działanie filtrów i innych aspektów GIMP-a, lepiej wykorzystując możliwości przetwarzania wieloprocessorowego Twojej karty graficznej. Bardziej szczegółowy opis Varuna można znaleźć w jego [raporcie GSoC](#).

Naprawdę doceniamy ciężką pracę naszych studentów GSoC!

Zespół projektowy

Jednym z obszarów, na którym chcemy się skupić po wersji 3.0, jest ulepszenie naszego procesu projektowania UI / UX . Utworzyliśmy [osobne repozytorium UX](#) , aby zgłaszać i omawiać problemy związane z projektowaniem. Chcemy zbudować zespół projektantów, aby omawiać i tworzyć ulepszenia projektowe dla GIMP , które również uwzględniają istniejące przepływy pracy użytkowników. **Denis Rangelov** bardzo zainteresował się tym obszarem i wykonał już świetną pracę, identyfikując, kategoryzując i przenosząc problemy projektowe z repozytorium kodu do dedykowanej sekcji projektowej.

Niektóre ulepszenia projektowe [zostały już zaimplementowane](#) w wersji 3.0 i nie możemy się doczekać współpracy z projektantami społeczności, aby zapewnić ludziom lepsze doświadczenia!



Co dalej?

Za kulisami dzieje się jeszcze wiele innych rzeczy i nie możemy się doczekać, aby podzielić się nimi z Wami wkrótce w poście z wiadomościami o wydaniu 3.0 RC1 ! Jeśli jeszcze tego nie zrobiliście, możecie przetestować wersję 2.99.18 ze strony pobierania wersji rozwojowych

<https://www.gimp.org/downloads/devel/>

Nie zawiera ona żadnych ulepszeń, które wprowadziliśmy od czasu jej wydania, ale nadal daje dobry podgląd tego, jak będzie wyglądać wersja 3.0.

Nie zapomnij, że możesz [przekazać darowiznę i osobiście wesprzeć finansowo deweloperów GIMP](#) , jako sposób na odwdzięczenie się i przyspieszenie rozwoju GIMP . Zaangażowanie społeczności pomaga projektowi stać się silniejszym!

Wersję PL przygotował:

Zbigniew Małach
Zbyma72age