Parallels-Utils

المحتوى

ضغط ونشر وإعادة استخدام صور Parallels الافتراضية عبر كل جهاز.

الملخص

لإدارة صور الآلات الافتراضية Server Factory هي مجموعة أدوات Parallels-Utils ضغط ومزامنة "مصفوفات" الصور المستخدمة في التطوير والاختبار: Parallels (نظام macOS)، ونشرها إلى نقطة نهاية بعيدة، واستردادها وتشغيلها على عدة محطات عمل أو خوادم. يمكن استخدامها بشكل مستقل أو كجزء من Server Factory.

وصف مختصر

مجموعة أدوات Shell/Python لإدارة دورة حياة صور Parallels الافتراضية على نظام macOS. تضغط هذه الأداة صور Parallels وتزامنها، وتنشرها إلى نقطة نهاية بعيدة، ثم تستردها وتشغلها على عدة حواسيب — مدفوعة بملفات إعدادات بسيطة، ويمكن استخدامها بشكل مستقل أو ضمن Server Factory.

وصف مفصل

تقدم Parallels-Utils حلاً عملياً لمشكلة DevOps في بيئات التطوير القائمة على macOS: أنظمة تشغيل وتكوينات مختلفة) Parallels حيث تبني الفرق "مصفوفات" من الآلات الافتراضية تُستخدم في التطوير والاختبار)، وتحتاج هذه الصور إلى ضغط ونشر واسترداد وتشغيل بشكل متسق عبر عدة أجهزة. توفر مجموعة الأدوات هذه الدورة الكاملة لإدارة الصور. آلية المزامنة تضغط صور Parallels وتحافظ على تزامنها؛ وآلية النشر تحمل الصور إلى نقطة نهاية بعيدة؛ وآلية الاسترداد تسمح لأي محطة عمل أو خادم بسحب الصور المنشورة وتشغيلها كآلات افتراضية. تم تصميم التكوين ليكون بسيطاً ومدفوعاً بالملفات: يحدد `image_location.settings` مكان وجود الصور في نظام الملفات، ويحدد نص `image_sync.sh` للصور المنشورة، ويحدد URL قاعدة `image_provider.settings`. يستخدم المشغلون البرنامج النصي للتحميل — مع أمثلة مقدمة في مجلد `Examples`. تتطلب الأداة `run.sh` للنشر و `publish_images.sh` لتشغيل الآلات الافتراضية Parallels. تم تصميم مجموعة الأدوات للاستخدام المزدوج: Python 3 وmacOS للإصدار المناسب من Server Factory يمكنها العمل كجزء من مشروع أكبر أو بشكل مستقل تماماً، بما يعكس فلسفة الفصل في المؤسسة. كما أنها تأتي مع رابط فيديو تعليمي قصير. كجزء من عائلة Server-Factory، تكمل الأداة Qemu-Utils (المكافئ لنظام Linux/QEMU)، مما يمنح النظام البيئي إدارة صور الآلات الافتراضية على كل من منصتي macOS/Parallels وخوادم متعددة المنصات/QEMU.

لماذا قمنا ببنائها

مشاركة بيئات التطوير والاختبار الافتراضية المتسقة عبر الفريق أمر ممل — فالصور كبيرة، وكل جهاز بأتمتة عمليات الضغط والنشر والاسترداد بحيث Parallels-Utils يحتاج إلى نفس المصفوفة. تقوم الافتراضية قابلة للتكرار في أي مكان Parallels تصبح مجموعة معيارية من صور.

لماذا تُعد نقلة نوعية

الثقيلة وغير العملية إلى مجموعة من القطع القابلة للنشر والمزامنة يمكن لأي Parallels تحول صور جهاز سحبها وتشغيلها — بحيث تتوقف بيئة التطوير والاختبار المعيارية عن كونها شيئاً يعيد كل مهندس بناؤه يدوياً لتصبح شيئاً يمكنك جلبه ببساطة. تفعل ذلك باستخدام ملفات إعدادات بسيطة للغاية وبدون محافظةً على فلسفة الفصل في المؤسسة: مفيدة بحد ،Server Factory أي اعتماد على بقية ذاتها، ومواطنة جيدة في سلسلة الأدوات الأكبر.

المحتوى

ما الجديد في الابتكار

- لأغراض التطوير والاختبار Parallels ضغط ومزامنة "مصفوفات" صور.
- سير عمل النشر والاسترجاع لجعل الصور قابلة لإعادة الاستخدام عبر أجهزة متعددة.
- ضبط الإعدادات عبر ملفات الإعدادات (الموقع/المزود/المزامنة) مع أمثلة مرفقة.
- استخدام مزدوج: إما بشكل مستقل أو كعنصر من عناصر Server Factory.

التحديات والحلول

- **توزيع الصور الكبيرة:** تم حلها عبر الضغط بالإضافة إلى سير عمل النشر إلى نقطة نهاية بعيدة والاسترجاع منها.
- **قابلية التكرار عبر الأجهزة:** تم حلها عبر إعدادات المزود والموقع لضمان أن كل مضيف يحل نفس مجموعة الصور.
- **سهولة الاستخدام:** تم حلها عبر نصوص برمجية بسيطة مثل `run.sh` و`publish_images.sh` وملفات إعدادات أمثلة.

مجموعة التقنيات (السبب والاستخدام)

- **Shell** — نصوص برمجية للنشر والتشغيل والمزامنة (اللغة الأساسية، ~5.3 ألف بايت).
- **Python 3** — أدوات داعمة (اعتماد ضروري، ~3 آلاف بايت).
- **Parallels (macOS نظام)** — الخلفية الافتراضية التي تتم إدارتها.
- **ملفات الإعدادات (`.settings`)** — ضبط تصريحي للموقع والمزود والمزامنة.

> ملاحظة: يشير GitHub إلى أن المستودع هو نسخة فرعية ضمن منظمة Server-Factory. متخصص لنظام macOS وغير مرتبط بـ AI.