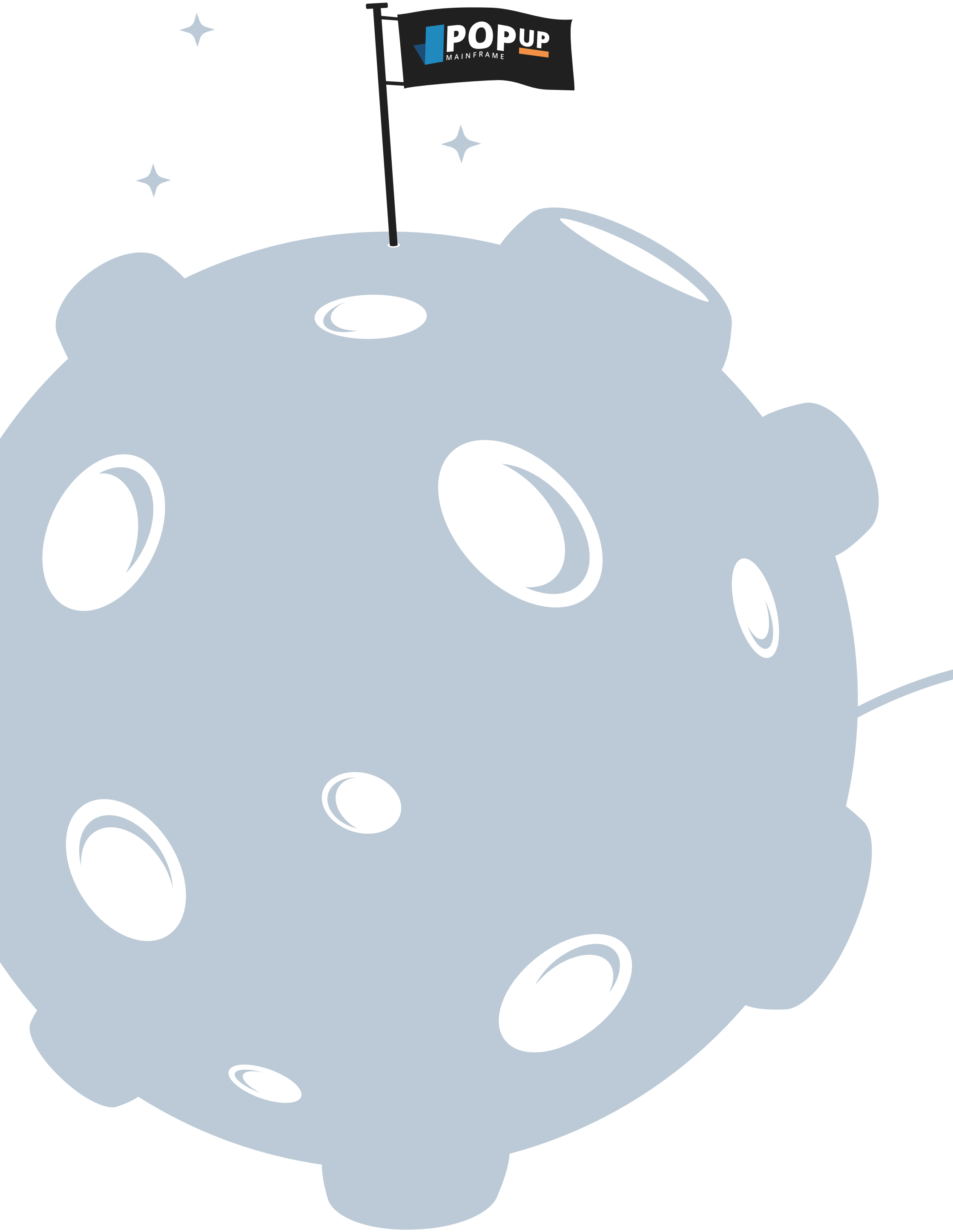
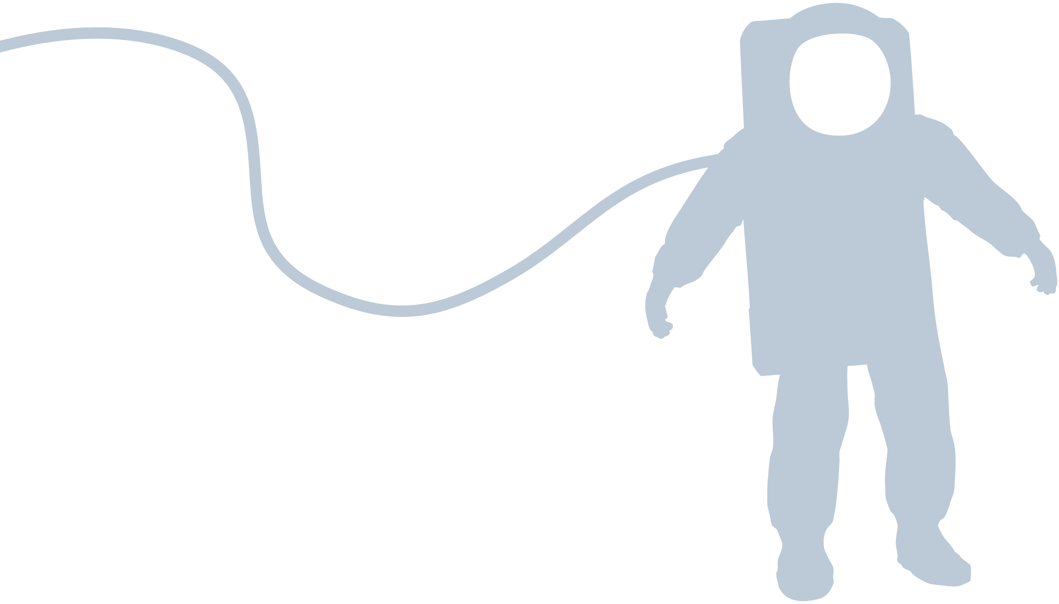
HANDBOOK


POPUP
MAINFRAME

CONTENTS

1. Welcome to the Future of Mainframe!
2. PopUp: Mainframe On-Demand
3. Your Mainframe Modernisation Strategy: Streamlined & Agile
4. Use Cases
5. PopUp FAQs



1. WELCOME TO THE FUTURE OF MAINFRAME!

Why Modernise your Mainframe?

Mainframe change can no longer be a bottleneck for organisational change. If it's holding you back, the time to act is now!

It is vital for organisations to keep their mainframe applications up-to-date because mainframe is still a critical part of many companies' IT infrastructure. By evolving their mainframe applications, businesses can improve their customer service, performance, security, and compliance.

What's the Urgency?

Why fix something which isn't broken? Well, in many cases the mainframe change processes are not fit for purpose anymore – they are no longer boosting a business' competitiveness, they are hindering it. Agility, speed and DevOps are key aspirations for nearly all application areas, but mainframe is falling behind other areas.

The recent 2023 BMC Mainframe Survey tells us that investment in mainframe is growing across industries (63% of survey respondents reporting an increase in mainframe investment over the past year). Enhancing automation and cost optimisation were listed as a top 4 priority for more than 46% of extra-large mainframe shops (more than 50k MIPS).

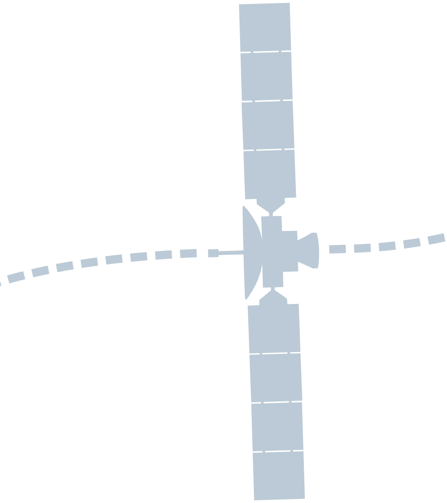
Investment in modernising the mainframe is growing.
Do you want to be left behind?

Mainframe Change is Tough!

Evolving the way change happens in mainframe is tough – and for good reason. Teams have typically been operating for decades, managing business-critical systems and responding to regulatory requirements with strict governance processes and deeply ingrained working patterns.

Your hurdles to mainframe application and software change might be...

- Modernising my mainframe is too hard – where do I start?
- My mainframe system is too complex and not well documented, so we don't touch it unless absolutely necessary.
- My senior mainframe resources are all busy on mission-critical work. I can't pull them away to come up with ideas, work on proposals and wait for approvals.
- I need to grow my team before I can take on a project like this.
- My budgets won't stretch to the specialised skills and tools needed for mainframe modernisation.
- I can't risk downtime on my business-critical mainframe systems.
- Changing my mainframe applications would require thorough regression testing which is not possible with our current test packs.
- My team is happy with the status quo. Asking them to change their ways of working and collaborate with other teams will meet a lot of resistance.



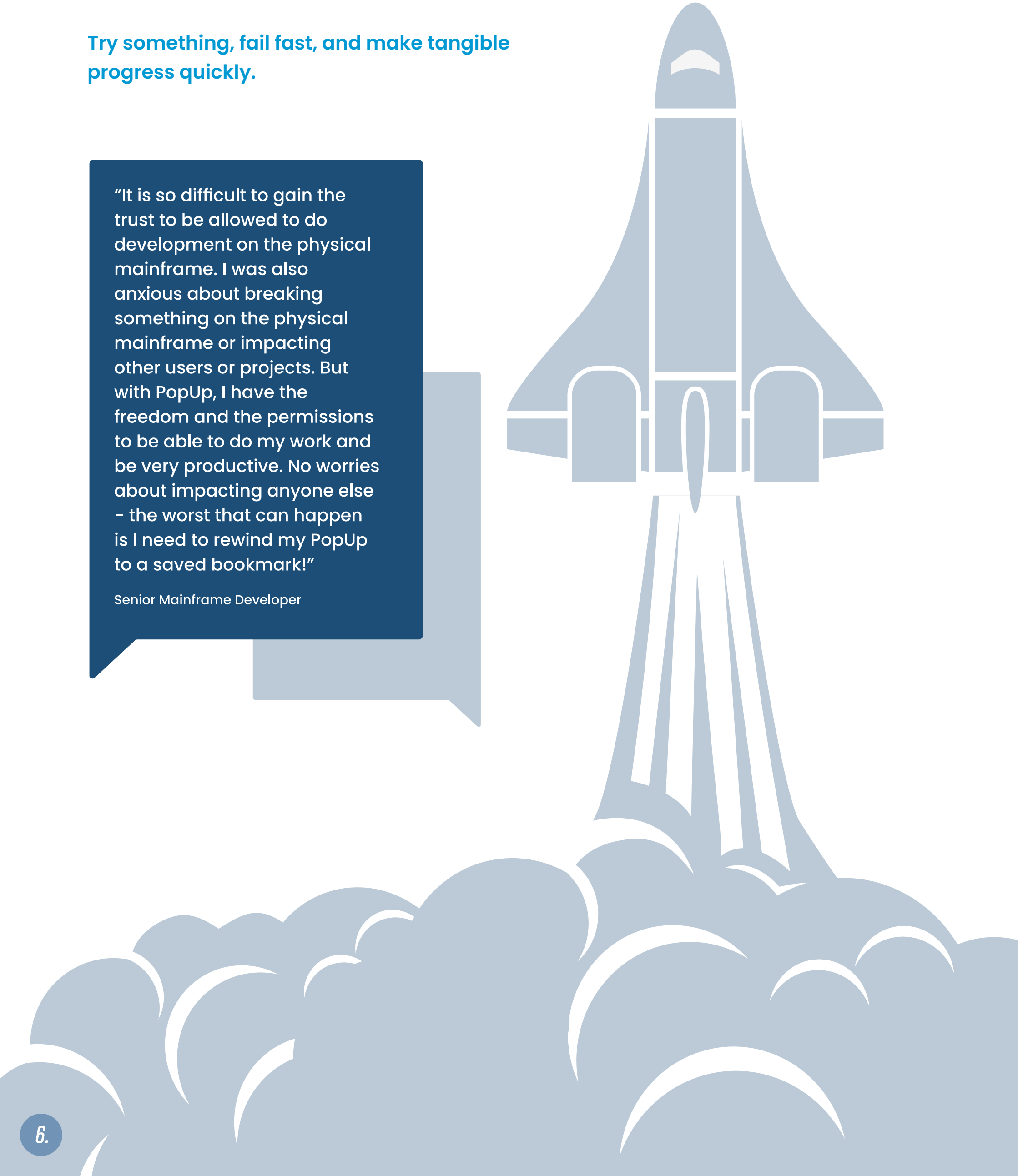
> "PopUp Mainframe is one of the most innovative software start-ups to have emerged in the last decade. Their unique blend of software and skills is catalysing mainframe transformation initiatives on a multitude of fronts, ranging from DevSecOps to cloud-native z/OS experiences. Their 'secret sauce' is an ability to vastly simplify what has historically been ultra-complex. I've seen them provision full-stack z/OS environments in Azure in a matter of hours - and I understand they are now able to do this in minutes."
>
> PopUp Business Partner

Start Small

Why not start your mainframe transformation journey with a small, low cost and low risk project? Avoid sucking up lots of time with your expensive senior people debating options and waiting for approvals.

Try something, fail fast, and make tangible progress quickly.

> “It is so difficult to gain the trust to be allowed to do development on the physical mainframe. I was also anxious about breaking something on the physical mainframe or impacting other users or projects. But with PopUp, I have the freedom and the permissions to be able to do my work and be very productive. No worries about impacting anyone else - the worst that can happen is I need to rewind my PopUp to a saved bookmark!”
>
> Senior Mainframe Developer

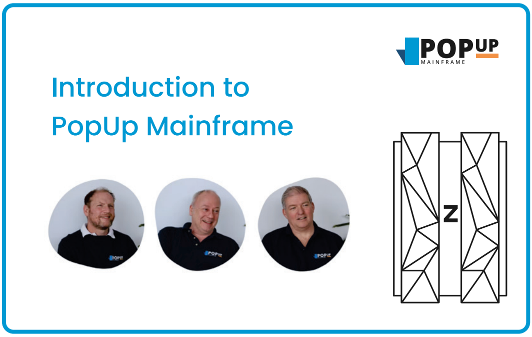


2. POPUP: MAINFRAME ON-DEMAND

PopUp Mainframe was born from a simple idea: what if we could make deploying real mainframe capability as simple as spinning up a virtual machine image on any x86 machine?

Well, it took some truly concerted effort and a lot of work but that's exactly what we have built.

Find out more in our video: Introduction to PopUp Mainframe



Introduction to PopUp Mainframe

The PopUp Ethos

PopUp enables you to eat the elephant of mainframe modernisation

PopUp Mainframe provides on-demand environments for Dev and Test with zero impact or risk to your physical mainframe. It facilitates mainframe transformation by making it easy for organisations to get started.

With PopUp, you can have a slice of mainframe functionality to fit your requirement at a fraction of the cost. This commoditises mainframe by enabling each person to have their own private mainframe instance and maximises their productivity.

This is the simplest way to accelerate time to market.

Our Vision

PopUp's vision is to empower businesses with modern ways of working so they can deliver frictionless mainframe change at the speed of the business.

All mainframe organisations have a common problem: The shortage of non-prod environments for development and test. PopUp directly addresses this with instantly available, fully functioning mainframes.

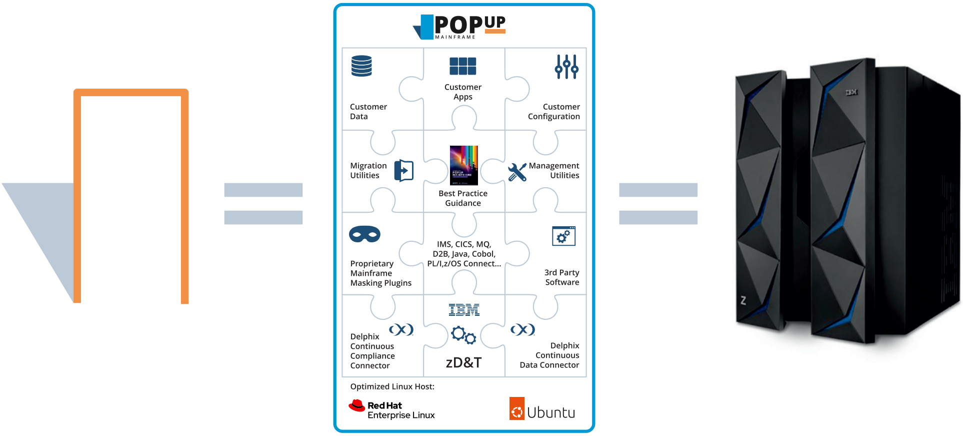
PopUp:

- Provides dedicated mainframe instances running on x86 hardware which can be torn down and recreated as needed - completely risk free to your physical mainframe.
- Liberates people to use modern mainframe and DevOps tools outside of the stringent governance process for physical mainframe.
- Comes with modern mainframe tools and utilities pre-installed to accelerate developer adoption of productivity-increasing tools e.g. IDEs, automation tools.
- Enables process and culture change in a zero-risk environment.
- With Delphix Continuous Data configured, PopUp can be quickly rewound to facilitate test reruns.

> "Our organisation tried to stand up ZD&T. We spent months working on it and gave up in the end. Once we had PopUp, it was deployed in only 2 hours and we had a virtual mainframe ready to use!"
>
> PopUp Mainframe Customer

What is PopUp?

A PopUp Mainframe image is a Linux appliance with IBM ZD&T pre-installed and pre-configured, as well as key IBM Enterprise software and PopUp helper utilities to simplify most admin tasks. A PopUp Mainframe is functionally equivalent to running a physical mainframe - it can be used in exactly the same way as a physical mainframe.*



What is PopUp?

The PopUp image can be run on any x86 architecture, be it on-prem or in the cloud. This provides a turnkey solution enabling rapid, elastic deployment of z/OS capability suitable for non-production use cases (including dev/test, training, issue replication & troubleshooting).*

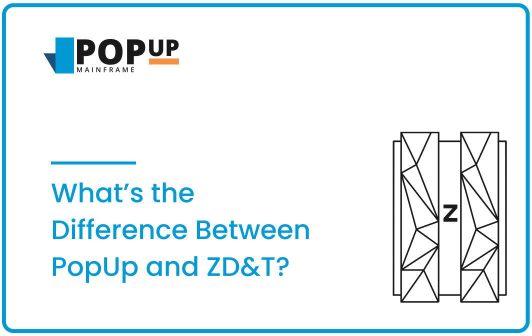
Your PopUp will look and behave exactly the same as your physical mainframe, as you can install any mainframe subsystem or bespoke application on your PopUp. We also provide migration utilities to simplify the movement of mainframe data to your PopUp.

*Note: the software license excludes performance testing and production workloads.

What is the Difference Between PopUp and ZD&T?

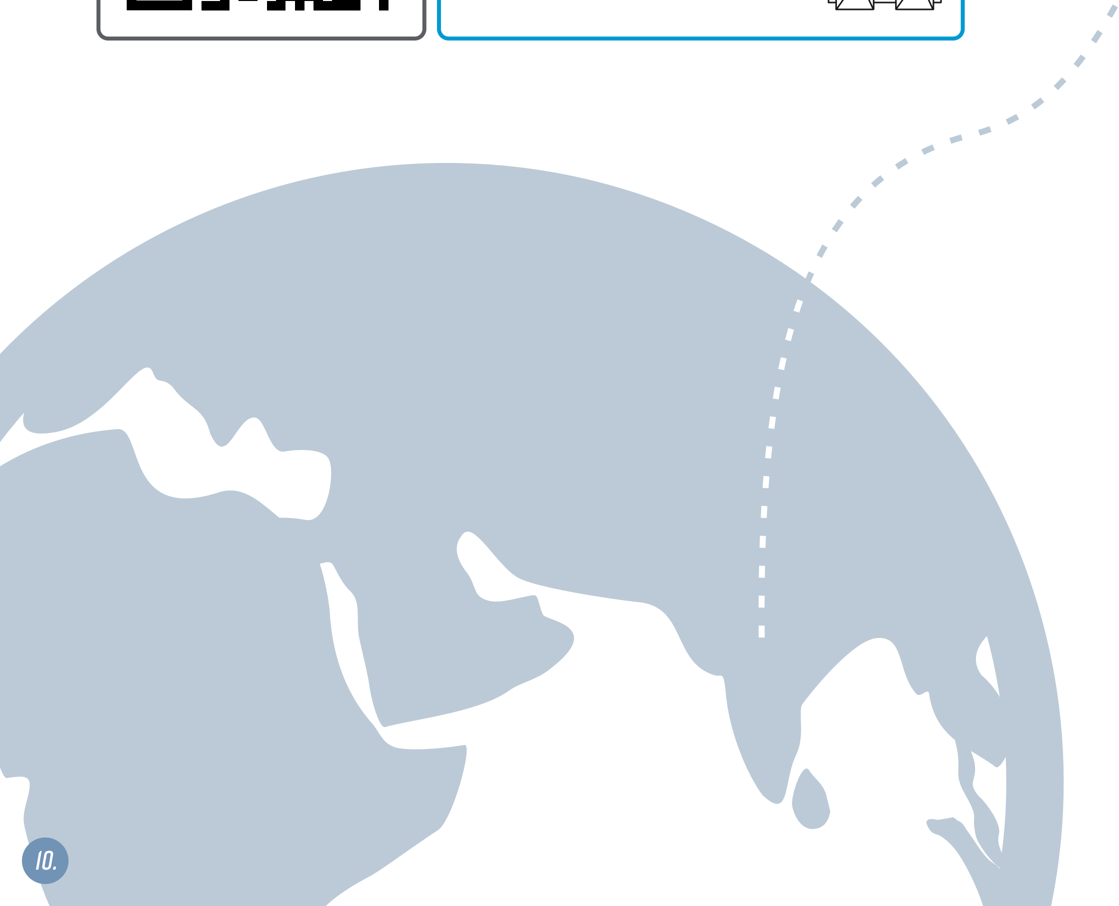
PopUp Mainframe is built on IBM ZD&T software. We’ve used our extensive expertise to optimise ZD&T, develop helper utilities, and package it up to make your adoption of ZD&T as quick and painless as possible.

Find out more in our video: What’s the Difference Between PopUp and ZD&T?



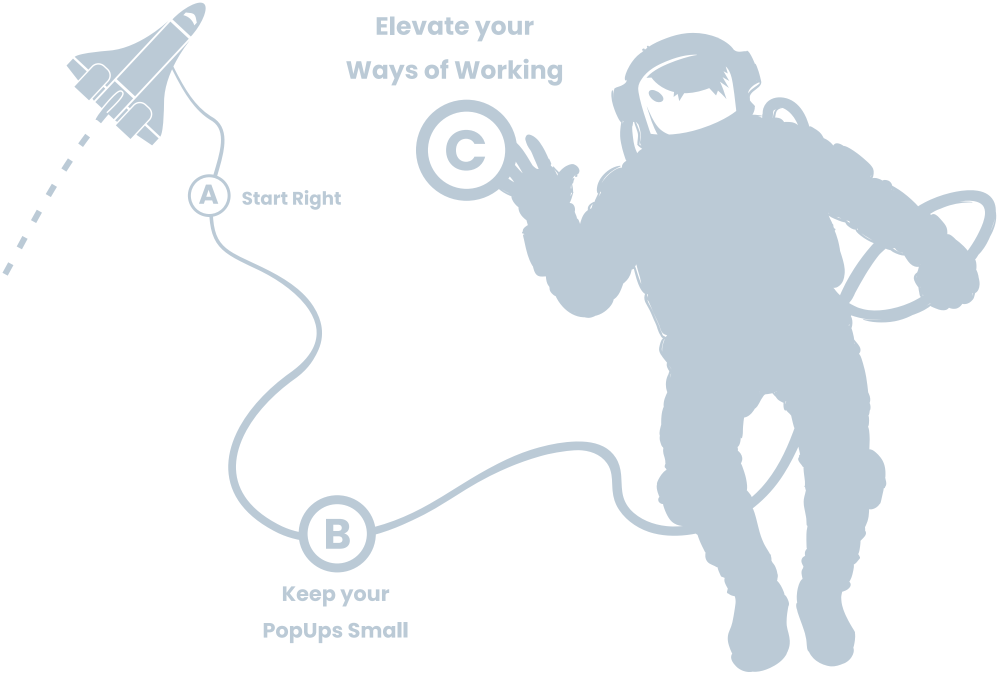
POPUP MAINFRAME

What’s the Difference Between PopUp and ZD&T?



3. YOUR MAINFRAME MODERNISATION STRATEGY: STREAMLINED & AGILE

These best practices come from our experience supporting a range of businesses on their mainframe transformation journeys. They will help you to get the most out of your PopUp.



A. Start Right

1. **Ensure you have the right people** with the transformation vision, modernization skills and senior stakeholder buy-in to move your organisation forward. They will instil the right attitude so that real change happens, and sticks.
2. **Encourage fast adoption of PopUp and new mainframe tools:**
   a. Which challenges will PopUp address in your organisation?
   b. How will the dev and test teams' work become easier?

B. Keep your PopUps Small and Lightweight

3. **Build new mainframe instances with just the essential application components for each use case.** Don't migrate your entire LPAR! A small PopUp footprint reduces overheads and gives more flexibility and control, e.g. easily allocate more resources (memory or CPU) for great performance. Example: To improve the performance of a Db2 application, tune just the Db2 app on the PopUp, then release the validated code and config changes to all environments. A full application ecosystem is not needed on the PopUp.
4. **Restrict third party software on your PopUp.** If you have a utility on an LPAR why not continue to use that and migrate data sets to PopUp using an IDE or the Migration Utility? Enable developers to use their favourite tools on the physical mainframe and keep your PopUp footprint small.
5. **Plan for each PopUp instance to be used by one person.** As PopUp users often have full permissions, people sharing a PopUp are likely to impact each other, e.g. during testing.
6. **Keep security simple.** Don't implement RACF. Give full autonomy to your dev and test teams to make changes on the PopUp to increase productivity and reduce dependence on sysadmins. In the isolated PopUp there is no need for tight security as there is zero risk to the physical mainframe.
7. **Keep maintenance simple.** Utilise PopUp (and Delphix) out of the box features for easy maintenance:
    a. Use each PopUp for a short period of time only, then tear it down.
    b. Use PopUp utilities to monitor processes and automate housekeeping e.g. JES logs.
    c. When building an application on the PopUp, implement the existing application housekeeping along with it. Don't reinvent the wheel!

C. Elevate your Ways of Working

8. **Utilise modern mainframe tools which come with PopUp** to increase productivity. For example, modern IDEs enable anyone to develop and test mainframe changes and increase the efficiency of the team.
9. **Implement automation.** PopUp comes pre-installed with automation tools including (amongst others) IBM DBB (Dependency Based Build) for build automation, and Topaz Total Test for test automation.
10. **Shift left your testing.** Test earlier and in parallel using multiple PopUps to accelerate projects (see Chapter 4).
11. **Consider using Delphix** if you plan to provision, clone and rewind PopUps regularly. With Delphix, you can create virtual PopUps with are truly 4-dimensional: checkpoint, rewind and fast forward your mainframe instance to a point in time.

4. USE CASES

There are many valid ways to use a PopUp.
Below are some of the highest impact use cases.

Shift Left your Testing to Accelerate Projects

1. I need to accelerate projects but I don't know how
2. I want to use cloud compute for mainframe testing

Commoditise Mainframe Environments for All

5. I need to hire new team members and get them skilled up quickly
6. I need an easier way to manage and refresh my test environments

Accelerate Batch Testing

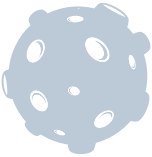
9. Batch testing is too painful and slow
10. I want to be able to test my batch in a more efficient way

Increase Agility & Deliver Change Faster

3. So much time is spent on manual activities, there is no time for anything else
4. I need to speed up and streamline the way we deploy changes

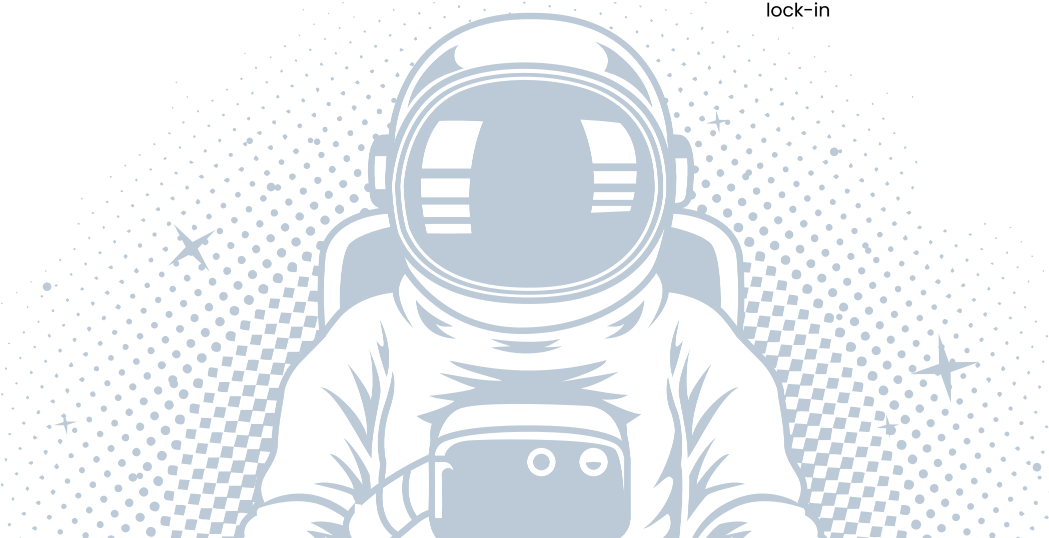
Upgrade with Ease

7. I want to take advantage of new features in the latest versions of my applications and subsystems
8. I can't upgrade z/OS as it's too risky



Break up your Monolith

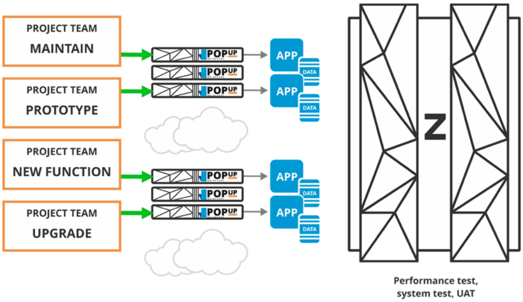
11. The mainframe needs to evolve at the same pace as other systems
12. I need a multi-cloud strategy to avoid vendor lock-in



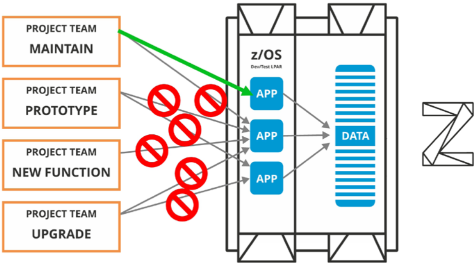
USE CASE I.

I need to accelerate projects but I don't know how

Is your testing constrained by the unavailability of mainframe environments? Can only one person or team access the mainframe at any one time?



With PopUp, virtual mainframe instances can be spawned on-demand on x86 machines. This completely removes contention on the physical mainframe.



All testing levels can be performed on the PopUp (unit, system, integration, UAT etc.) by integrating with other systems. Testing activities can be done earlier and concurrently - shifting left the entire SDLC.

To **shift left** testing with PopUp, simply:

1. Stand up a PopUp either on-prem or in the cloud (a plain PopUp or cloned from a PopUp 'Gold Copy' containing pre-configured applications and data).
2. If using a plain PopUp:
   - migrate only the necessary mainframe applications onto the PopUp
   - Export data from production
   - mask the data
   - copy it to the PopUp
   - Save this as a PopUp Gold Copy for next time
3. Use the PopUp for development and testing.
4. One complete, tear down the PopUp.
5. Perform UAT and release to production.

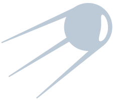
Use Delphix Continuous Compliance to provide fully masked data to dev and test environments so people can always test with 'real' data.

USE CASE 2.

I want to use cloud compute for mainframe testing

Using PopUp, you can run your mainframe instances on prem or in the **cloud**. Using PopUp in the cloud gives you a pay-what-you-use mainframe compute capability. PopUp is available to purchase directly through the Azure Marketplace, and the PopUp image can be deployed in any other cloud.

Just like on-prem, PopUp in the cloud supports all levels of mainframe testing.



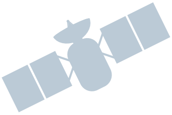
INCREASE AGILITY & DELIVER CHANGE FASTER

USE CASE 3.

So much time is spent on manual activities, there is no time for anything else

Bringing **automation** to your SDLC is a sure-fire way to accelerate change and increase the productivity of your teams. Implementing automation on the PopUp is simple as each PopUp instance comes with automation tools and utilities pre-installed, and you avoid the red tape approvals processes required on the physical mainframe. For example, Topaz Total Test can be shipped pre-installed on the PopUp, and it enables users to build automated test packs and more.

Give your teams the freedom and flexibility to be **innovative:** Use PopUps as disposable sandbox environments where people can experiment with new ideas, proof of concepts and prototypes with zero risk.



USE CASE 4.

I need to speed up and streamline the way we deploy changes

Implement an **end-to-end pipeline** to automate control of mainframe change. Using pipelines, you can spin up a mainframe instance, deploy your code, run tests, then tear down your instance – all on-demand.

Take this opportunity to modernise your mainframe governance model and streamline approvals by mandating automated quality checks and approvals in the pipeline. Watch your change velocity go up and up!

COMMODITISE MAINFRAME ENVIRONMENTS FOR ALL

USE CASE 5.

I need to hire new team members and get them skilled up quickly

Gone are the days when green screen is the only option for mainframe development. Attract new people to join your team and empower juniors by offering intuitive, easy-to-use IDEs and self-service portals. Modern mainframe tools provide fast feedback and make automation easy to implement. PopUp ships with modern tools available out of the box.

Provide indestructible mainframe instances for training students and employees alike. You have full control over the training environments (configuration, setup, data) and they pose zero risk to the physical mainframe.

USE CASE 6.

I need an easier way to manage and refresh my test environments

Using PopUp in conjunction with Delphix you can achieve 4-dimensional mainframes - the power to create, clone and rewind mainframe environments at will.

Using PopUp and the time travel capabilities of Delphix Continuous Data, you can:

- Virtualise a PopUp instance and clone it in minutes. Once you have your PopUp Gold Copy, provisioning a clone to any users who need it is quick and effortless. The only limit is your imagination!
- Bookmark a PopUp to save a specific moment in time. Bookmark at key moments like just before testing, or once data has been loaded.
- Say goodbye to housekeeping and rewind or fast forward to a bookmark in just a few clicks! For example, bookmark before testing, run your tests then rewind to the bookmark and you have a fresh, clean copy of PopUp ready to go again if needed. No need to manually clear down the logs, databases or refresh test data.
- Provide self-service control to PopUp users through the Delphix online portal. Minimal Delphix skills are required to provision, bookmark, rewind and tear down PopUps.
- Provision and control PopUps through APIs.
- Reduce the storage footprint by 10x.

USE CASE 7.

I want to take advantage of new features in the latest versions of my applications and subsystems

Upgrading applications and subsystems can be an onerous task. You may be tempted to delay upgrades and avoid the perceived risk, time and effort involved. However, upgrades often come with enhanced features and capabilities which could benefit your business.

With PopUp, upgrading an application or subsystem is easy. Simply spin up a PopUp, install the new version, and test it with your application. New versions can be trialled and validated on the PopUp without any risk or impact to the physical mainframe and no cumbersome approval process/work order delays.

USE CASE 8.

I can't upgrade z/OS as it's too risky

Upgrading z/OS on a physical mainframe is risky and high impact and needs to be planned well in advance. Avoid all of this by using a PopUp instance to test out your apps and subsystems on a new version of z/OS. There is no need to make any changes to the physical mainframe. For example: build an application and set of test cases on a PopUp running z/OS 2.4, then migrate these to a PopUp running z/OS 2.5, and test.

PopUp currently supports z/OS 2.4 and 2.5.

> "As a junior you are very restricted in your learning because you are not allowed to experiment on the physical mainframe. The disposable PopUp training environment is amazing because I have unlimited access and I can work on my training with no risk to anyone else. I've learnt more about mainframe development using PopUp in 1 month than the previous 6 months of training!"
>
> Junior Mainframe Developer

ACCELERATE BATCH TESTING

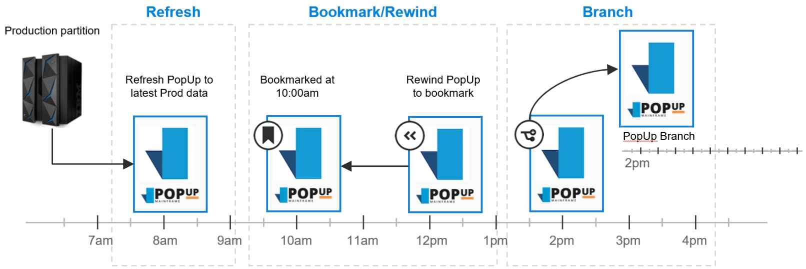
USE CASE 9.

Batch testing is too painful and slow

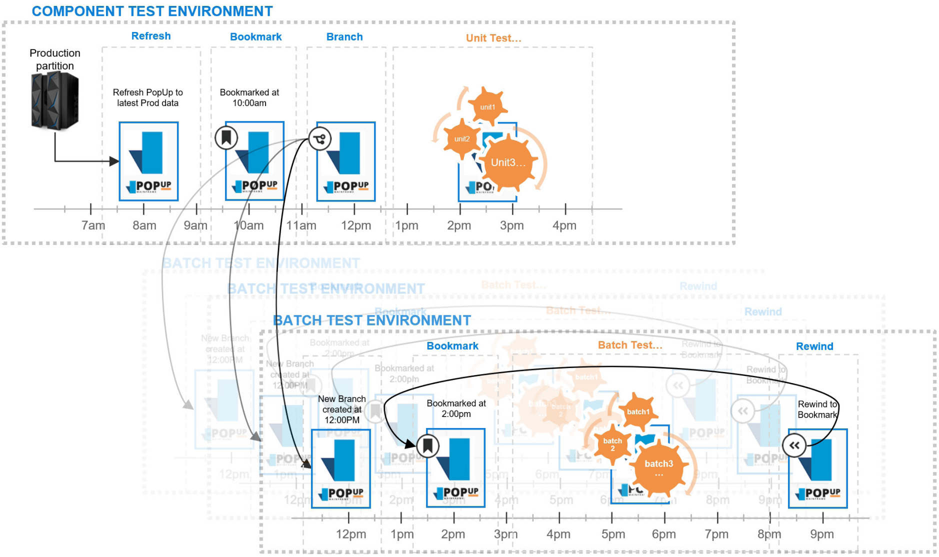
Is your batch testing performed at the end of a development cycle in a waterfall-like manner with high cost of fixes?

PopUp is a game changer for batch testing, enabling you to provision a 'clean', isolated mainframe instance which can be used to test your batch with zero dependencies.

Easily clone your PopUp to create a new instance, or use Delphix to branch your PopUp environment and test your batch in an isolated branch.



Using branching, multiple batch runs can happen in parallel, supporting different streams or other systems simultaneously.

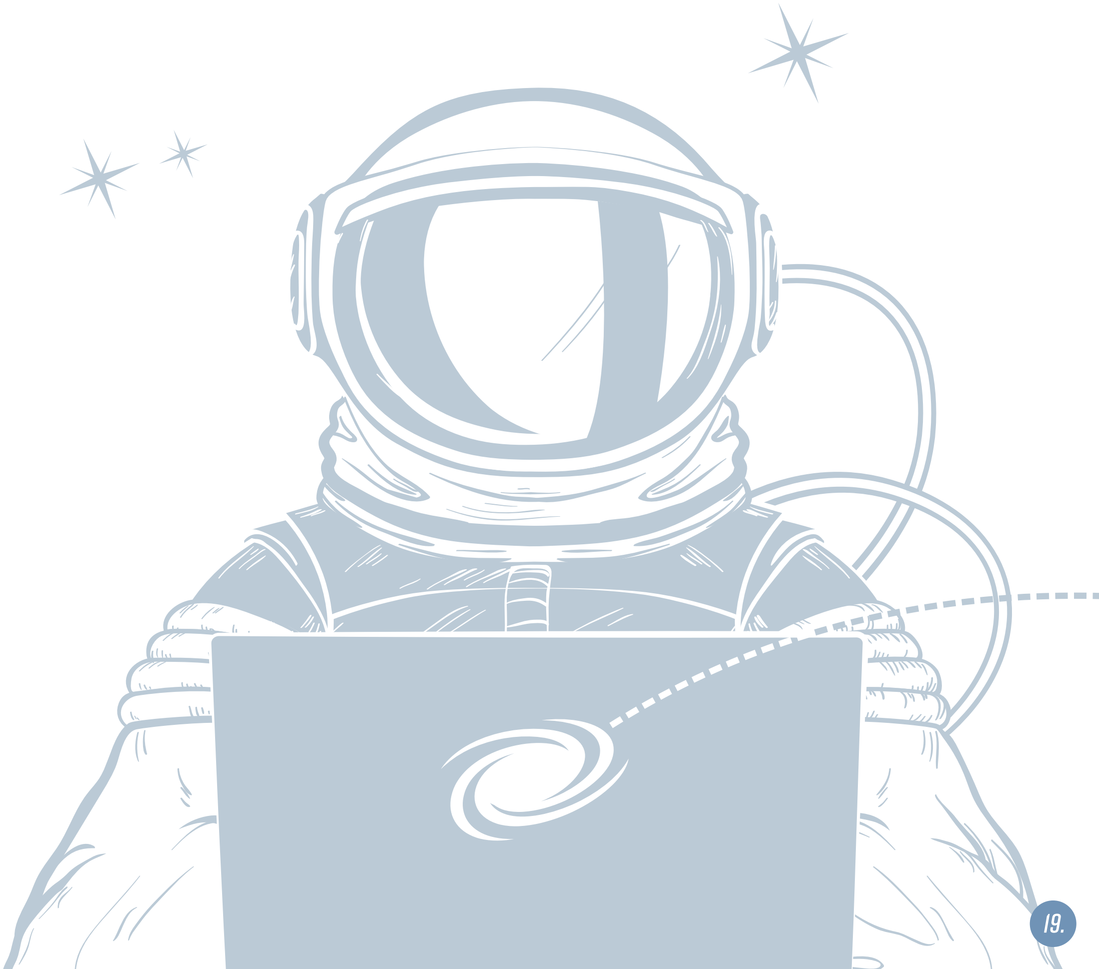


USE CASE 10.

I want to be able to test my batch in a more efficient way

Using the power of Delphix with PopUp Mainframe, navigate your monthly or quarterly batch cycles to test intra-batch code changes immediately. Use Delphix bookmark, rewind and forward capabilities to step back and forward in your batch run timeline as required. This game changing approach combined with some deft pipeline orchestration (employing parameterized test variables to manage input & output values and maintain appropriate test validation rules), facilitates mid batch code changes without invalidating or re-starting the entire batch run.

So, with a DevOps pipeline and orchestration tooling, test only what needs to be tested, and streamline your batch timeline to ensure deadlines are always met. At PopUp we call this “Tardis Testing”… as it provides a vehicle for cheating time and becoming a testing Time Lord!



BREAK UP YOUR MONOLITH

USE CASE II.

The mainframe needs to evolve at the same pace as other systems

Within large organizations, a historical divide often perpetuates between mainframe and distributed development teams. The adoption of automated pipelines and “DevOps initiatives” on the mainframe tend to significantly lag the implementation of such strategies at the distributed tier. This is perhaps the biggest reason for application changes on the mainframe being perceived as slow and hard. It may be tempting to avoid making change on the mainframe, and instead “go around” wherever possible by building new functionality in distributed systems.

But this is a mistake… and it can rapidly snowball into a mountain of unnecessary complexity and technical debt.

PopUp Mainframe bridges the gap and makes it possible to safely experiment with bringing DevOps practices to the mainframe at the speed of business, with zero risk to existing physical mainframe environments. With this freedom, it’s just as quick to refactor mainframe applications on the PopUp, keeping the application functionality close to the systems of record, increasing performance whilst reducing complexity and unnecessary compute power.

Test out new tools and automation pipelines in a controlled manner within a sandboxed PopUp instance and say goodbye to red tape, approval delay, security signoffs and waiting for a slot on the physical mainframe.

USE CASE 12.

I need a multi-cloud strategy to avoid vendor lock-in

In the realm of modern computing, whether working on premises or in the cloud, workloads can exist across hybrid cloud environments and diverse managed services. However, when attempting to transfer a workload or dataset from one cloud provider to another, you may encounter intricate obstacles often summarised with the term “vendor lock-in”.

PopUp Mainframe instances can be ported and copied easily between various cloud providers, avoiding vendor lock-in and providing your organization the option to pick and mix host environments which best suit your ever-changing business and budget requirements.

5. POPUP FAQS

How quickly can I setup a PopUp?

PopUp can be up and running in the cloud in 1 hour, and on-prem in under 2 hours.

Which software runs on PopUp?

PopUp supports z/OS 2.4 and 2.5 and **any** 3rd party software - including TSO/ISPF, Db2, CICS, IMS, MQ, and more. Future software and OS versions will be supported. PopUp is a fully provisioned mainframe and runs real z/OS code.

Can I connect PopUp to my physical mainframe?

Yes. You can copy application data, executables and source libraries onto the PopUp using an IBM utility, FTP or Connect Direct. You can also connect to your PopUp through an IDE e.g. Topaz.

What can I do on PopUp?

Anything that you might already do in Dev & Test on z/OS, including testing and development, R&D, third party software evaluation, training for testers and developers.

What CAN'T I do on PopUp?

PopUp cannot be used for production workloads including pre-prod testing and performance testing. Certain kinds of encryption are not available in PopUp.

How many PopUps can I have?

As many as you need. You can have one or multiple Linux servers / cloud instances running PopUp. You can save PopUp config, system and application data at any point, tear it down, then restore it at will.

How do I maintain my PopUp?

To keep maintenance to an absolute minimum, it is recommended you provision a "clean" / "Gold Copy" PopUp for each new use – this removes the need for long term maintenance. Your own z/OS sysadmin can resolve any ad hoc problems.

Do I need to have z/OS or green screen skills to use a PopUp?

No! In fact, PopUp enables developers with absolutely no mainframe experience to connect with, configure and develop z/OS systems with zero risk to the physical mainframe.

WHO ARE POPUP?

Mainframe Modernisation in Hyperdrive

Empowering mainframe businesses to deliver changes faster, cheaper and better with on-demand environments for Dev & Test

GARY THORNHILL

Founder and CEO

Guiding customers on their mainframe modernisation journey.

JOHN EDWARDS

Mainframe Lead

Advancing the PopUp strategy through his extensive technical experience.

ARUN MANDAPATI

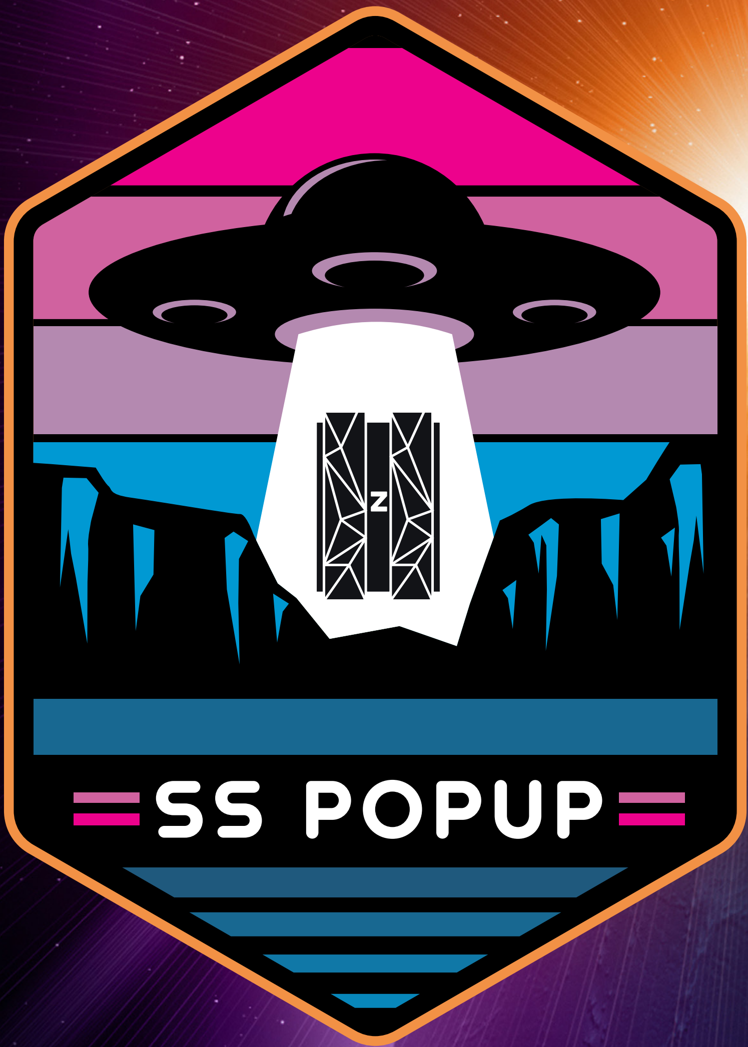
Engineering Lead

Providing the technical strategy and implementation of PopUp's Linux layer.

STUART FEASEY

Customer Solution Engineer

Helping customers bring mainframe development lifecycles into DevOps thinking.



If you are curious to find out more about how PopUp could help you in your mainframe modernisation journey,
contact us at:

popup-mainframe.com