Innovation Roadmap

# How to Get Innovation Unstuck and Solve Real Problems

Sooner or later, every agency that dreams of being “cloud smart” and innovative gets a sobering reality check. It turns out that moving applications to the cloud can be tougher than expected – and doesn’t necessarily lead to faster innovation. So, what’s the path ahead? Increasingly, agencies are adopting a rapid application development (RAD) approach. GovLoop, Red Hat and Emergent partnered to give you a look at the journey to innovation using this approach.

## Obstacles to Cloud-Smart Problem Solving

**Often organizations are not able to rapidly respond to new requirements because of both legacy systems – and legacy mindsets. Which of these following obstacles is your agency struggling with?**

### Concerns with moving legacy applications to the cloud, such as:

- Latency and other performance problems
- Lack of interoperability with other systems
- Dependencies on other on-premises systems

### Concerns with taking a lift-and-shift approach to cloud migration, such as:

- Lack of flexibility
- Lack of scalability
- Concerns about latency, performance issues
- Less ability to optimize cloud workloads

### Concerns about vendor lock-in, such as:

- Cost of transitioning from one provider to another
- Lack of technical expertise in transitioning
- Lack of portability of data and applications

# A Roadmap to Agility

**One of the first steps toward increasing your agility is to adopt a rapid application development methodology. RAD has four basic phases:**

1. **Requirements Planning:** Using brainstorming, user scenarios and other techniques to define broad requirements.

2. **User-driven design:** Taking an iterative approach to building prototypes, getting user feedback, building new protypes, getting more feedback and so on, until a vision takes shape.

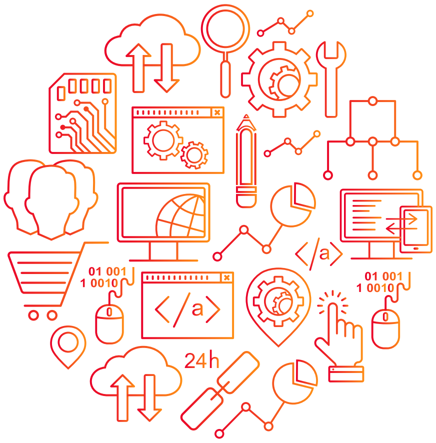
3. **Rapid construction:** Refining the prototypes and moving to application development and testing, while still getting user feedback.

4. **Cutover:** Testing the product, getting user sign-off and training users on the final product.

**True agility comes from the combination of a rapid application development methodology with cloud-native capabilities – including the use of microservices, containers and Kubernetes, for container orchestration.**

**Although it's possible to cobble together the tools you need for RAD, it pays to have those tools under one umbrella. Beyond, Kubernetes, that includes:**

- The services necessary to build and run containers from scratch
- A network to connect application services
- A load balancer to manage traffic coming into the Kubernetes cluster
- Persistent storage to back stateful application services
- Monitoring and logging to ensure the platform and applications run smoothly



# Best Practices in RAD

**To make rapid application development stick, consider these best practices:**

- **Practice planning, training and patience.** For example:
  - Take advantage of vendor resources to get developers and other staff on board and up to speed. Red Hat, for example, makes its Open Innovations Lab available to help agencies with transformation.
  - Invest in training to ensure seamless adoption. Once your team is ready to dive in, start with moving simple applications to containers.
  - Take the time to run the applications in parallel — the original legacy application on premise and the new containerized version in the cloud, which can help your team can learn as it goes.

- **Take a thoughtful approach to scaling.**
  - Remember this basic rule: The bigger the application, the more it has to be scaled for all of the services it's providing; consequently, the smaller the service, the more you can scale it where demand exists – instead of having to scale the full application. Take microservices to the extreme by scaling the unit size down to individual service levels.

- **Start with a trusted container base image to ensure security.**
  - A trusted container base image guarantees that you can run only images you trust in your container environment. Starting from a trusted base image means you will build secure applications from the ground up.
  - It's also a good idea to use separate build and runtime images, sticking to the restricted security context constraint where possible, and protecting the communication between application components using Transport Layer Security.

- **Find a strong partner.**
  - Few organizations have the knowledge and experience in-house to choose the right solutions and realize the full value of those investments. The key is finding a partner that works to understand your requirements, budget and long-term goals, and will stick with you from procurement, training and implementation through the long term.
  - Make sure your chosen partner has deep expertise in open source, cloud and platform technologies, and good relationships with vendors.

**Emergent, a trusted Red Hat public-sector specialist partner, is a solutions provider, systems integrator and value-added reseller that has provided solutions and guidance to federal agencies for nearly two decades. Emergent's 360 approach supports agencies at every stage of engagement, from architecting solutions and procurement to tailored delivery, training and ongoing support.**

**Learn more: www.emergent360.com/resources/red-hat-govloop/**