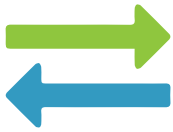


LEGACY SOFTWARE **EVOLVE** or **DIE!**



MIGRATING YOUR LEGACY APPLICATIONS MAKES SENSE

Legacy applications present a sustainability issue:

Maintainability and Reliability

Legacy applications are hard to refactor, hard to test and hard for new developers to wrap their heads around the code base.

Efficiency and Scalability

Many legacy applications consume more resources to maintain and ensure stability than do modern CFML applications. They also are prohibitive to lightweight, scalable microservices architecture.

Security

All applications are exposed to some risk, but legacy applications are significantly more vulnerable. Don't wait for the worst; act now.

Your Image

Software engineering is about modernizing, not staying 10 years behind.

Living in legacy land is like living in chains: you are restricted in all directions.

CHANGE YOUR MINDSET

Decide and commit to modernizing your application and stretch out of your "comfort" (discomfort) zone.

Embrace new development methodologies, tooling and techniques.

Evolve or die!





MODERN TOOLSET

Every migration requires the right tools and at Ortus we have the right modern toolset for any legacy migration.



COMMANDBOX



COLDBOX



CONTENTBOX



CFCONFIG



docker

ORTUS MIGRATION STRATEGY

- >> Create a holistic overview of what needs to happen with your team to modernize your legacy applications.
- >> Identify pain points in your application, which are candidates for extraction in to microservices.
- >> Create costs estimations and timeline.
- >> Execute your migration plan and don't look back.
- >> Use Agile to deliver iteratively.
- >> Secure support after project scope is completed with the appropriate SLA.



ORTUS FORMULA FOR SUCCESS



1 Identify performance bottlenecks and pain points in your application.



2 Build, test, and deploy new microservices to address and deliver the functionality of your pain points.



Update endpoints in monolithic app to use new microservice endpoints



Rinse and repeat, with additional bottlenecks and pain points