



Serverless Apps

Use Cases for Azure Functions

Author: Szymon Bernad

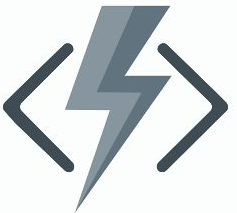


- Brief description of Azure Functions concepts
- Simple scenario - asynchronous request processing
 - Example built on .Net 7 Isolated Azure Functions
- Using Azure Functions to improve legacy web app reliability
 - Real-life case of a simple workaround



Azure... Functions?

- lightweight cloud services - part of MS Azure “serverless” platform
- offered in “consumption-based” or App Service Plan modes
- meant to have relatively short lifetime and be stateless by design

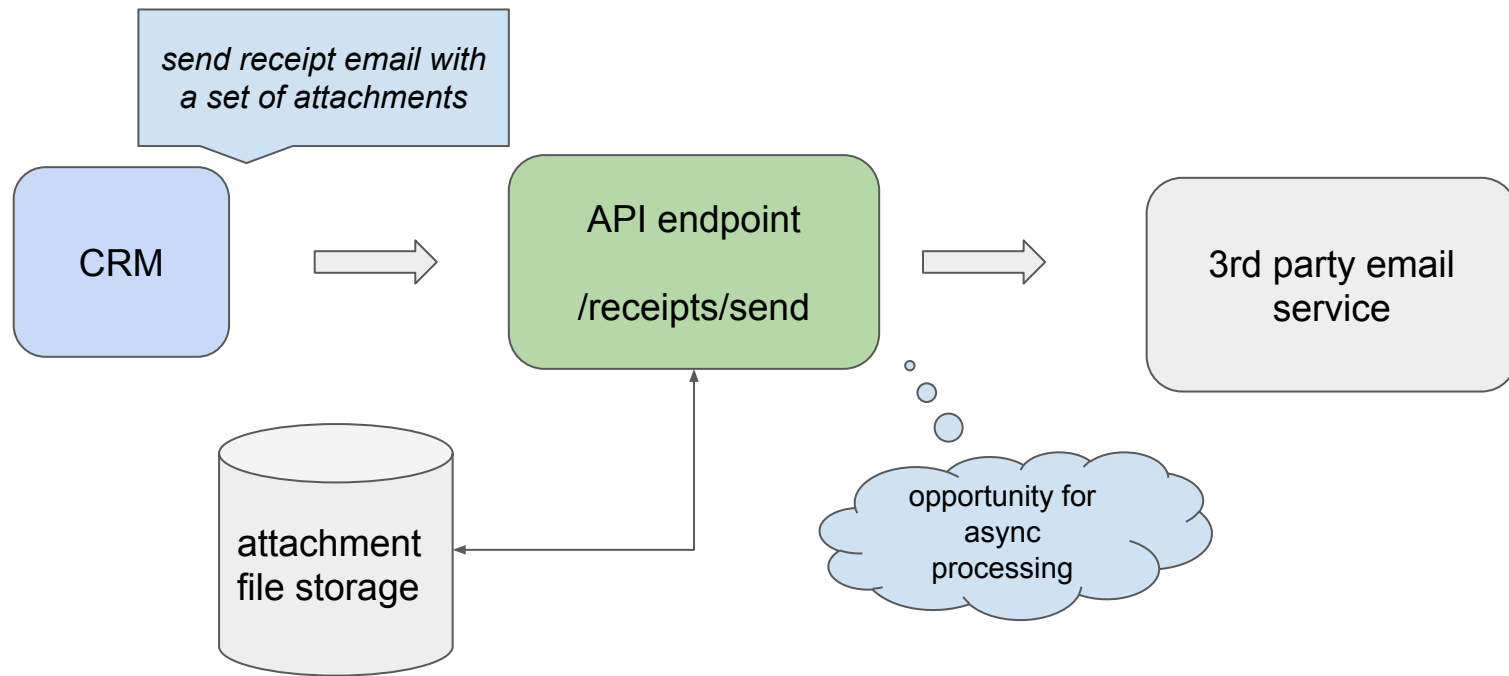


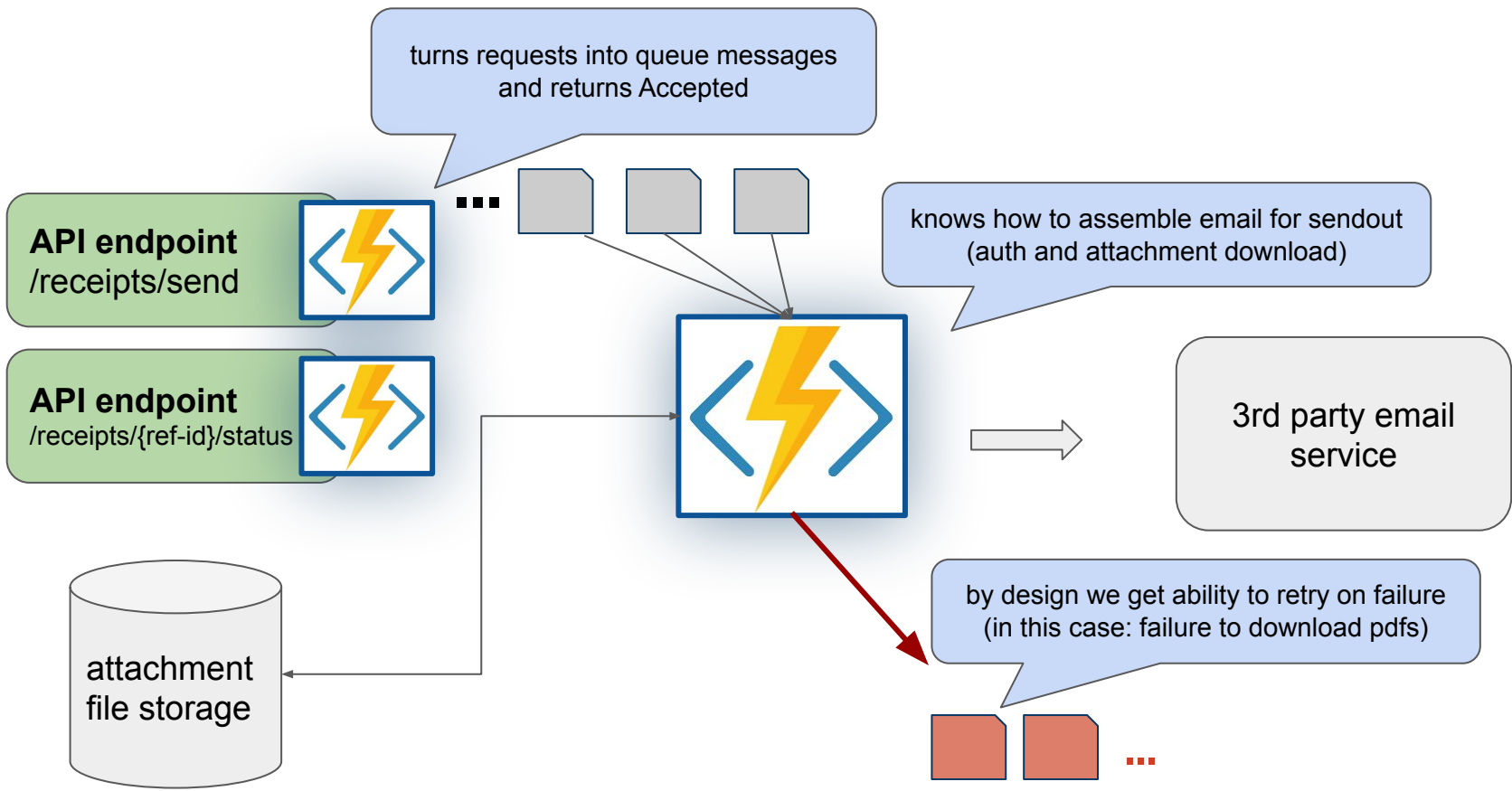
Azure... Functions?

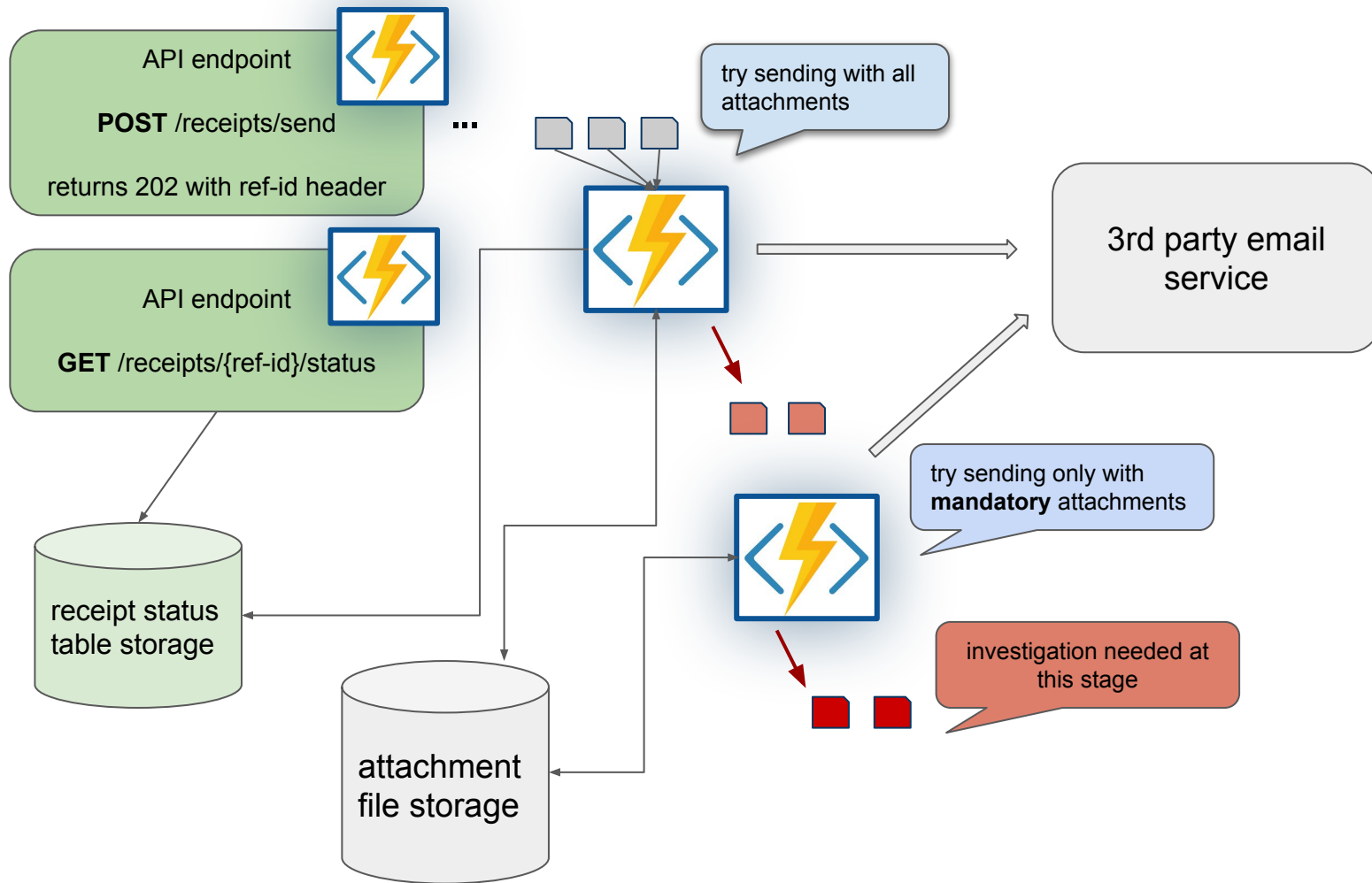
- many built-in integrations / input & output bindings
- improvements are still being added to the initial concept
 - Isolated vs In-Process (latest .Net, Asp.Net Core integration, Middleware)
[Differences between in-process and isolate worker process .NET Azure Functions | Microsoft Learn](#)
 - ReadyToRun binaries
 - Containerization with mcr.microsoft.com/azure-functions



Simple Use Case: asynchronous processing

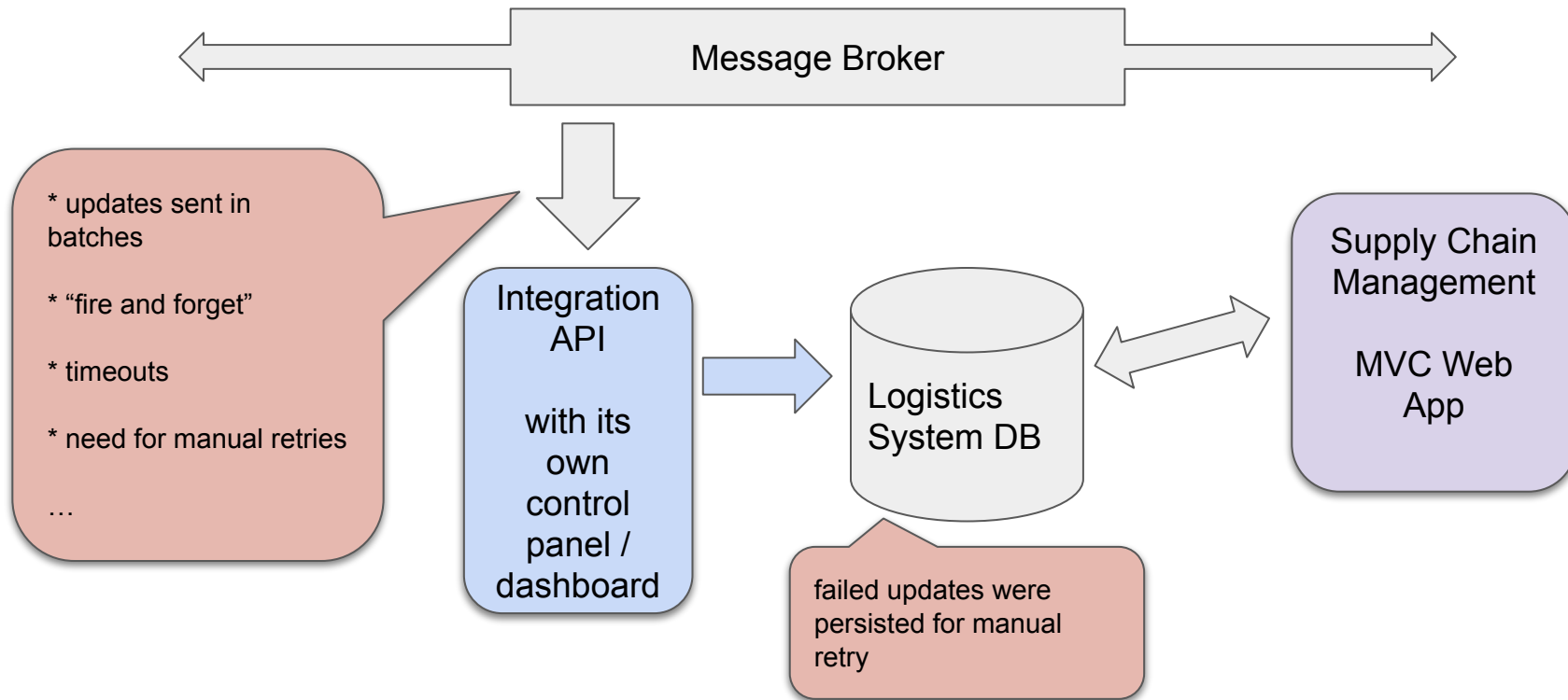






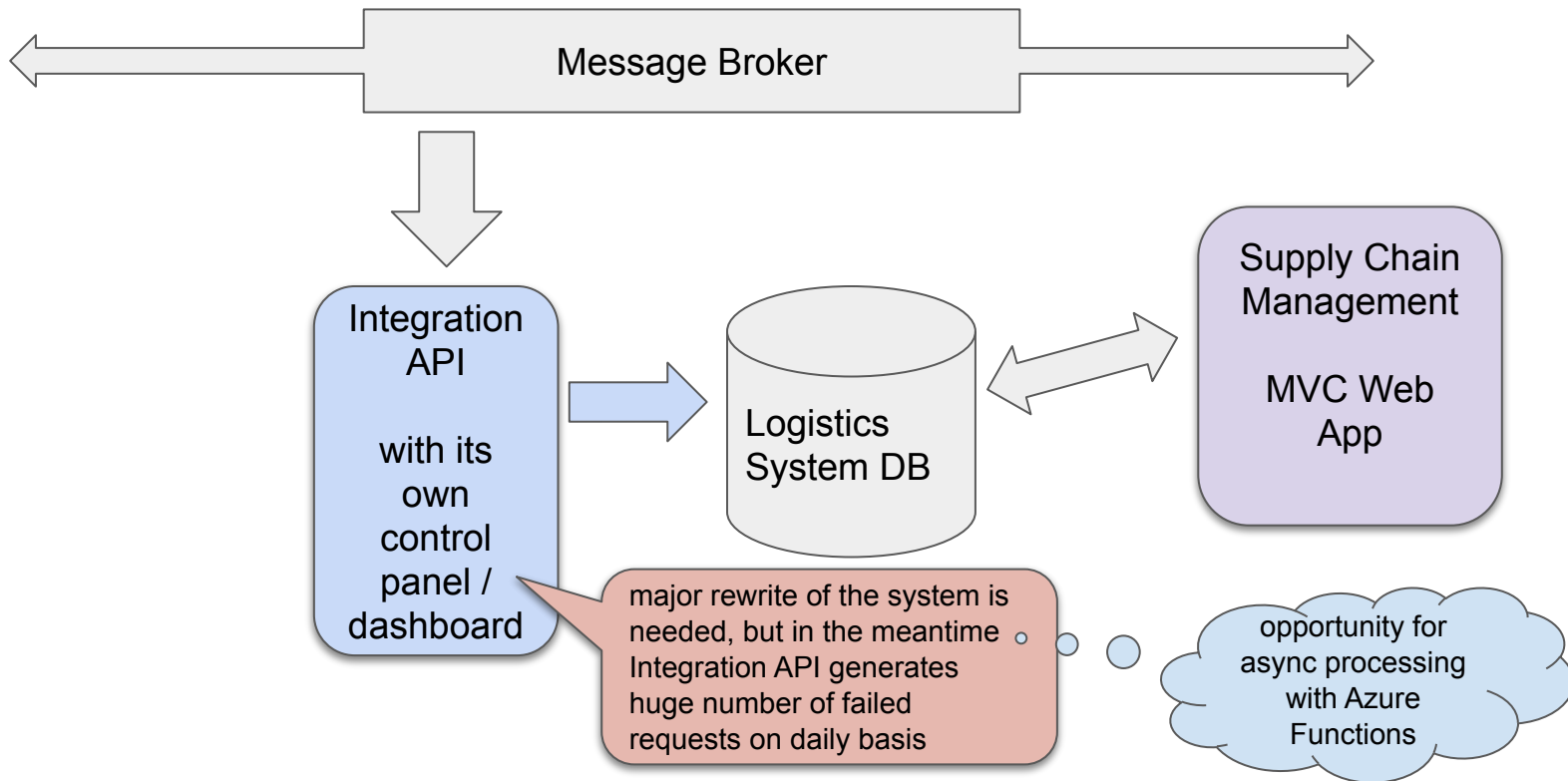


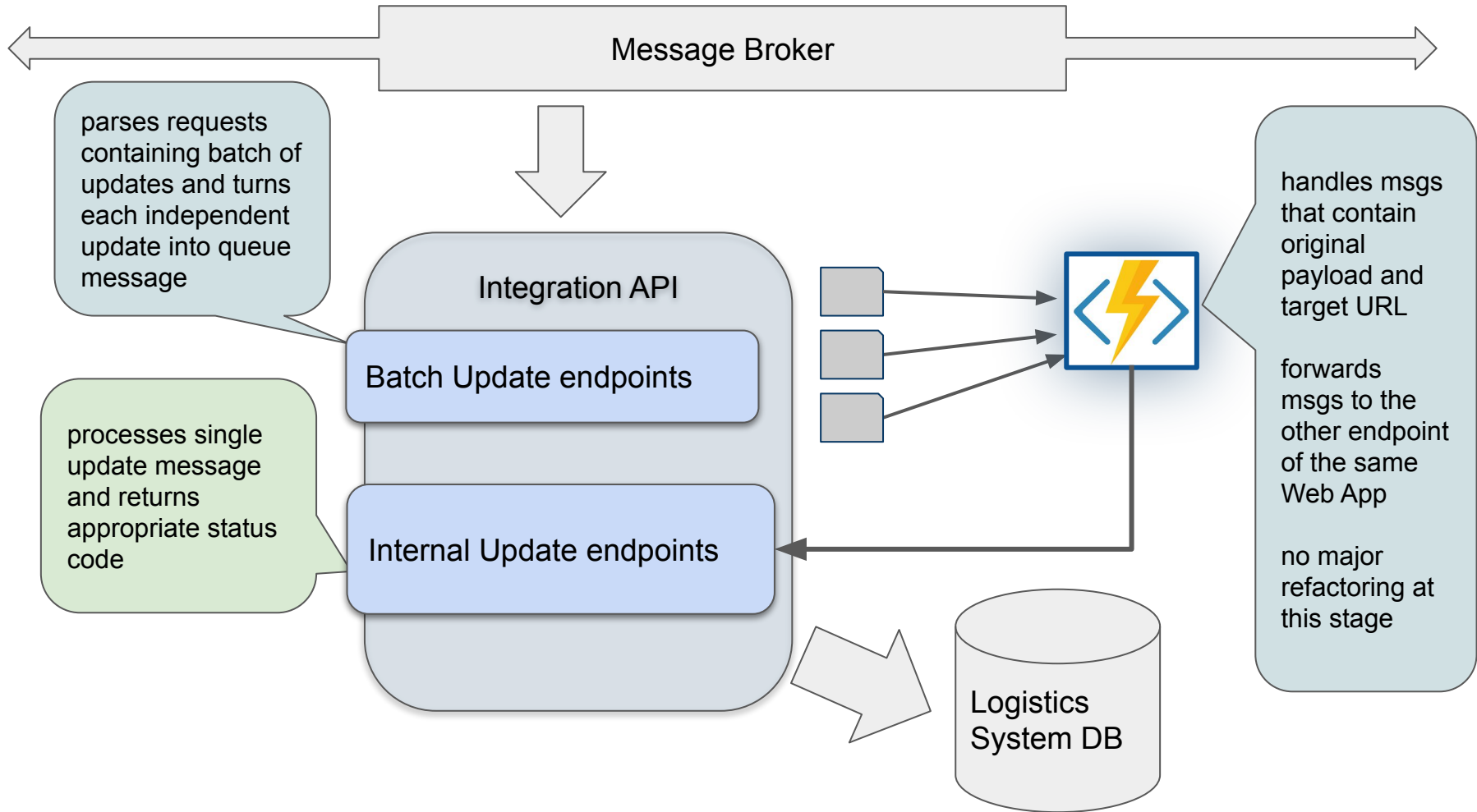
Using Azure Functions to improve legacy web app reliability





Using Azure Functions to improve legacy web app reliability







Using Azure Functions to improve legacy web app reliability

try

{



}

catch

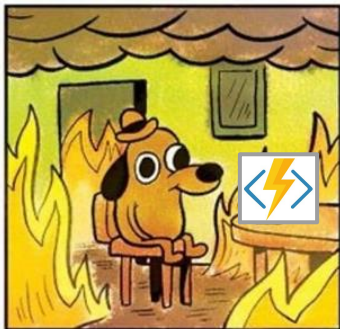
{



}

- interesting outcome of this workaround:
 - > 80% of update msgs succeeded on retry
- no need to change existing Azure infra as the same AppService Plan was reused
- opportunity to learn more about error patterns
- better use of ApplicationInsights logs
- increased confidence based on how easy it is to test & debug AzureFunctions

```
try  
{
```



```
}  
catch  
{
```



```
}
```

Thank you

[github.com/sim8500/
serverless-azfunc](https://github.com/sim8500/serverless-azfunc)