UC

Extending ArcGIS Enterprise with SOEs and SOIs

SHREYAS SHINDE

AKSHAY MALOO

CARSTEN PIEPEL

esri

Agenda

- Introduction
- Server object extensions
- Server object interceptors
- Esri consulting services projects
- Best practices & tips
- Q&A

Please help us improve this session by filling out the session surveys

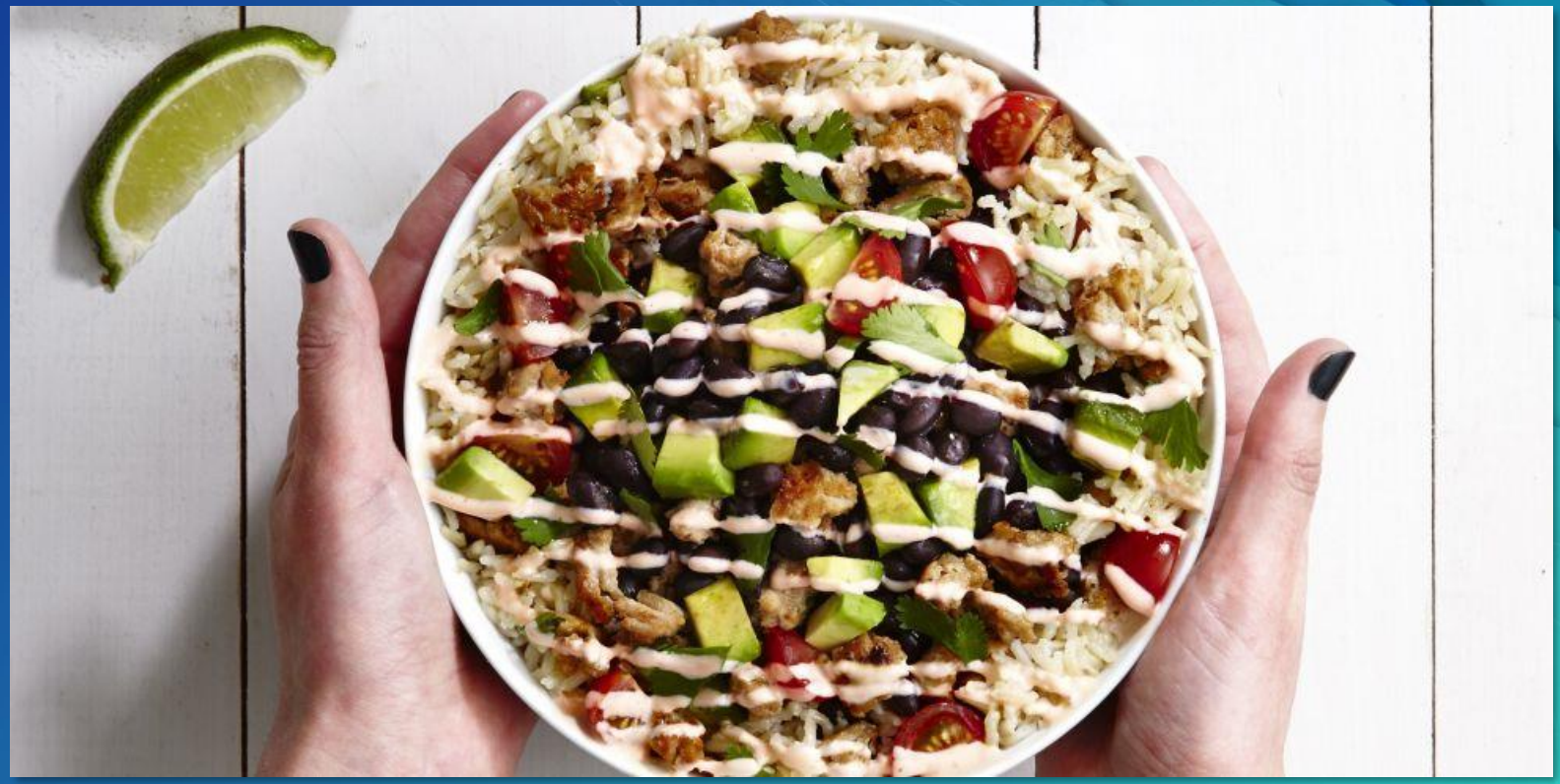
Introduction

ArcGIS Platform

- Powerful GIS toolset
- Building blocks
- Configurable and extensible

Why?

- Default is not adequate
- Custom business logic
- Security considerations
- Integrate disparate systems



> Because no two burrito bowls are the same

Options

Custom applications

Geoprocessing services

Server object extensions

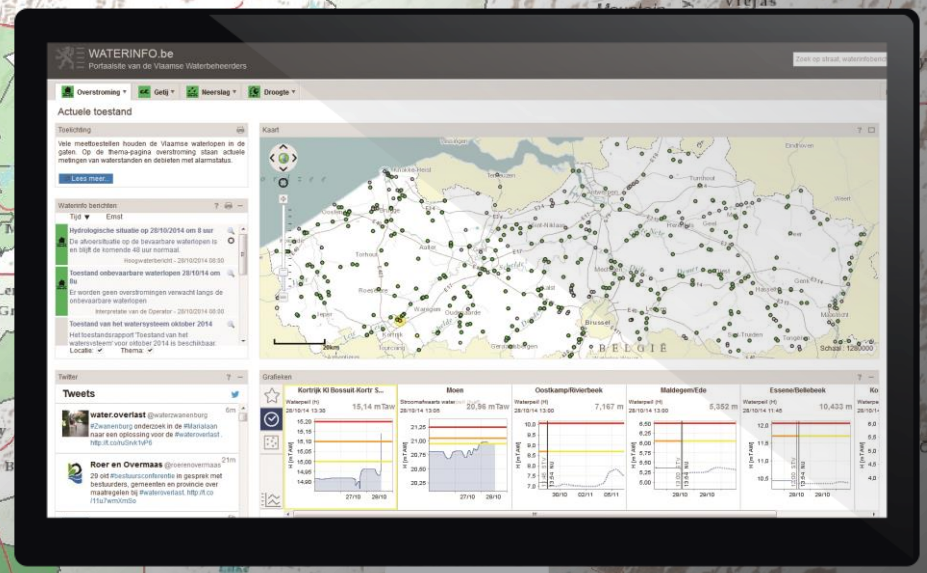
Server object interceptors

Options

Custom applications

Business logic coded in your application

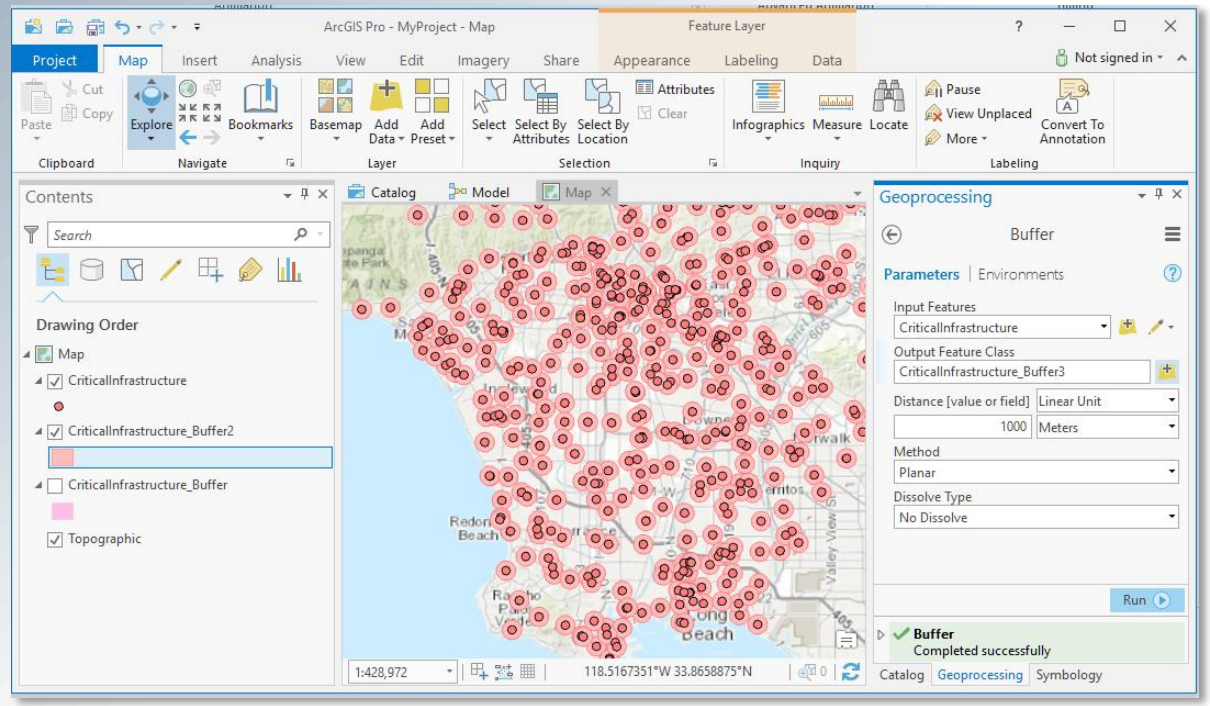
- Performance
- Reusability
- Scalability

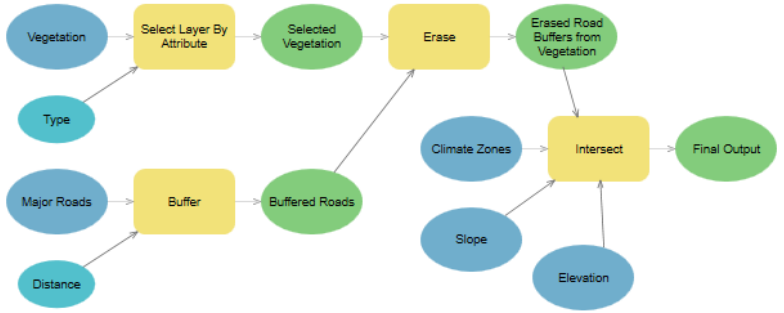


Options

Geoprocessing services

- Full GIS library
- Asynchronous
- Python





> Bread, butter for a GIS analyst

Server Object Extensions

Server-side code

Attached to Map or Image services in GIS Server

Access to full suite of arcobjects

Custom API (REST | SOAP)

Performance

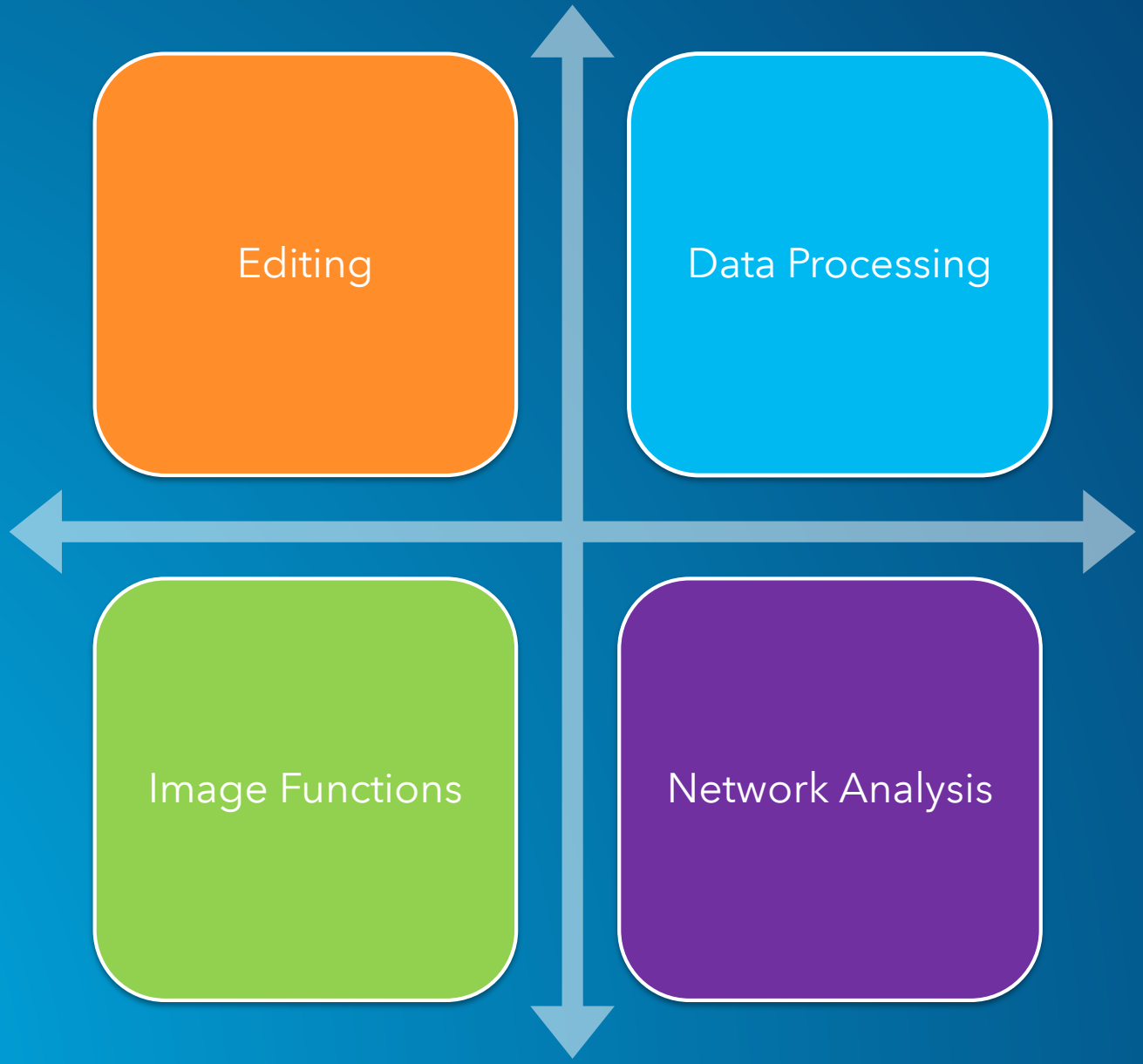
Managed lifecycle

No additional hardware

Authentication and authorization

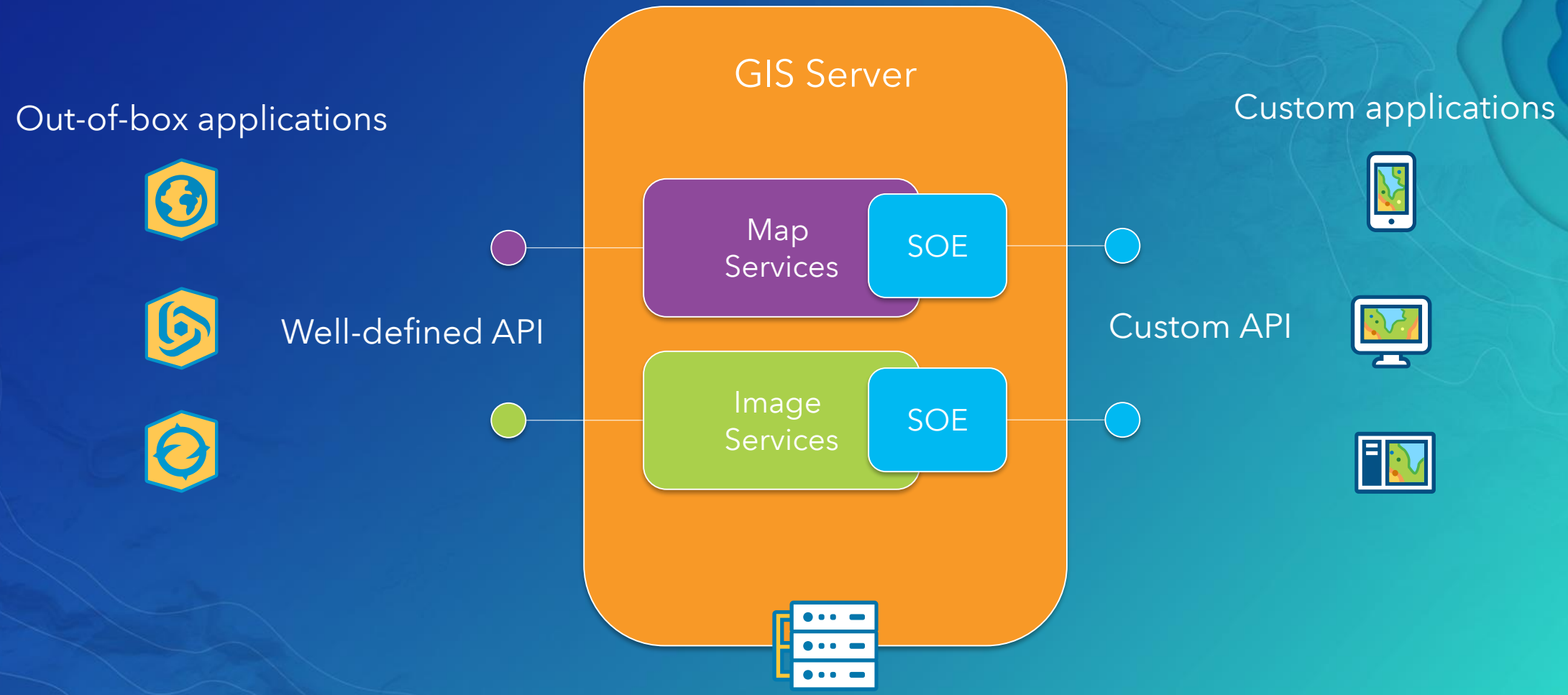
Allows you to focus on your business logic!

Use cases

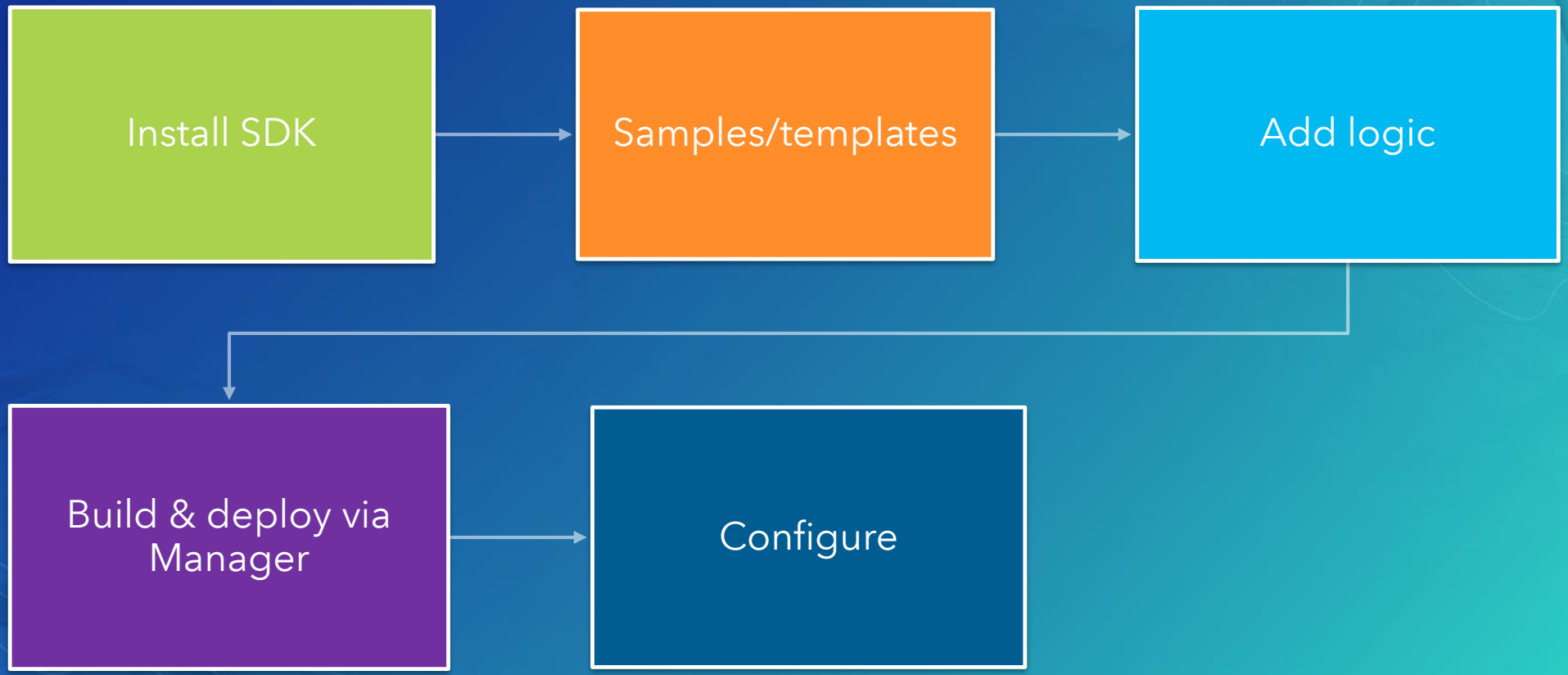


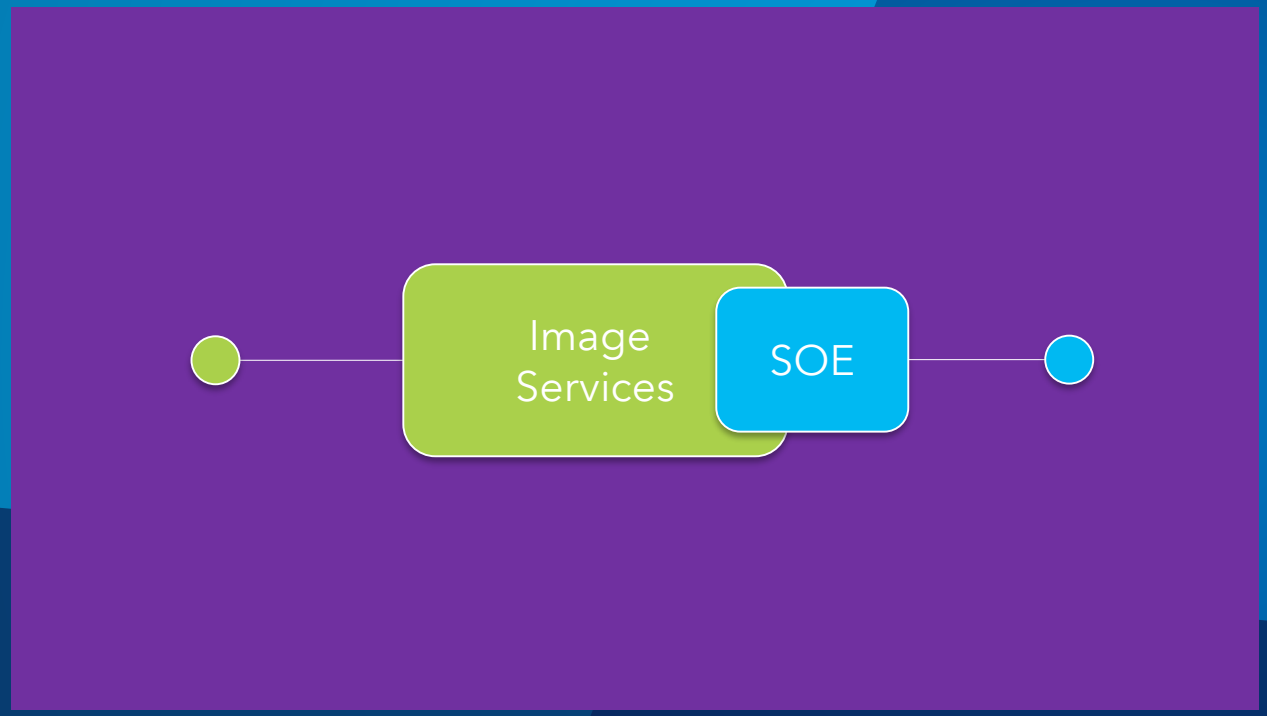
When performance is key!

Architecture



Process


Install SDK
Samples/templates
Add logic
Build & deploy via Manager
Configure



Demo

Server Object Extensions

Server Object Interceptors

Server-side code

Map or Image services

Filter requests and responses

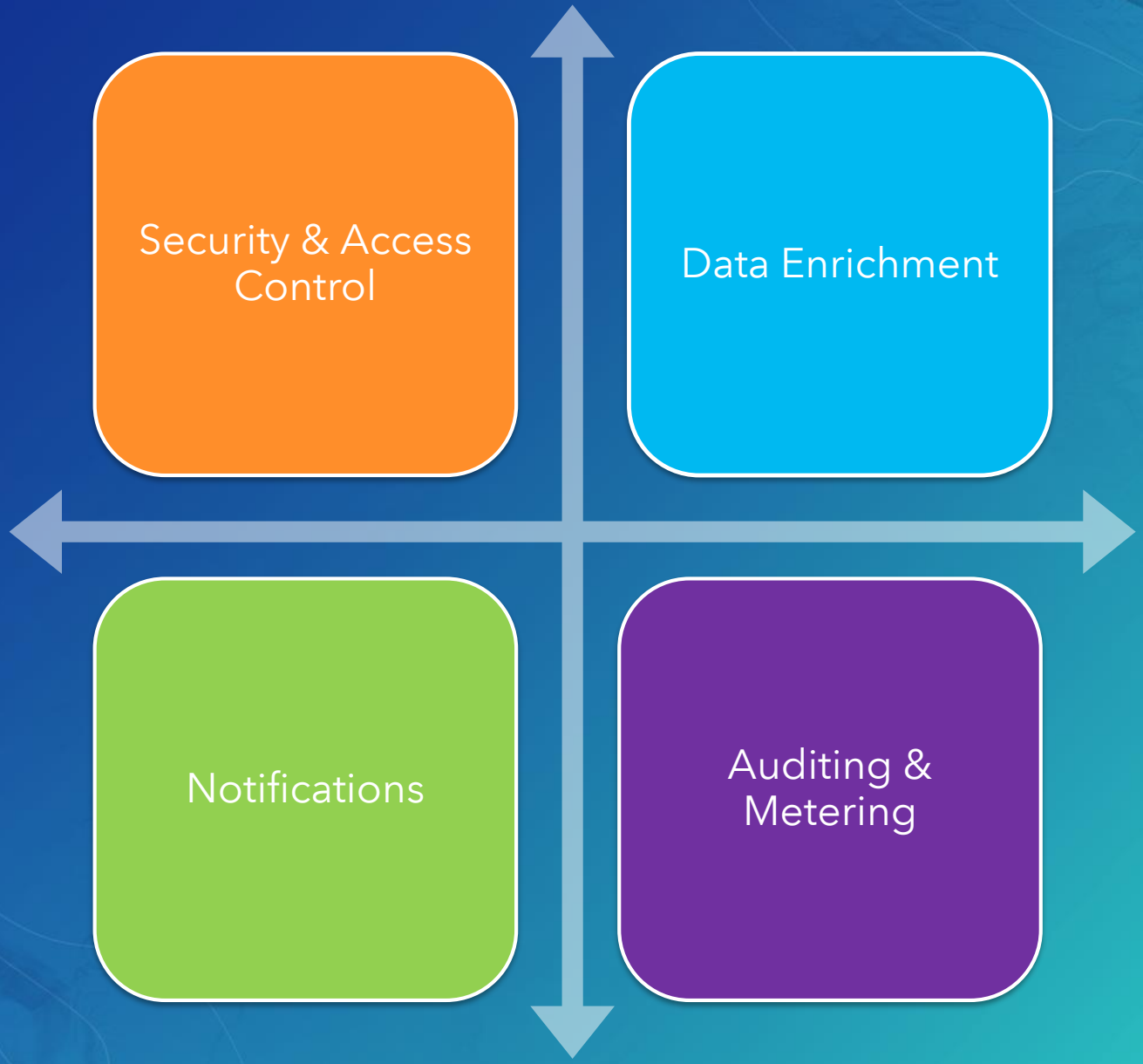
Chains

API cannot be changed

Must handle all map/image API

Managed lifecycle

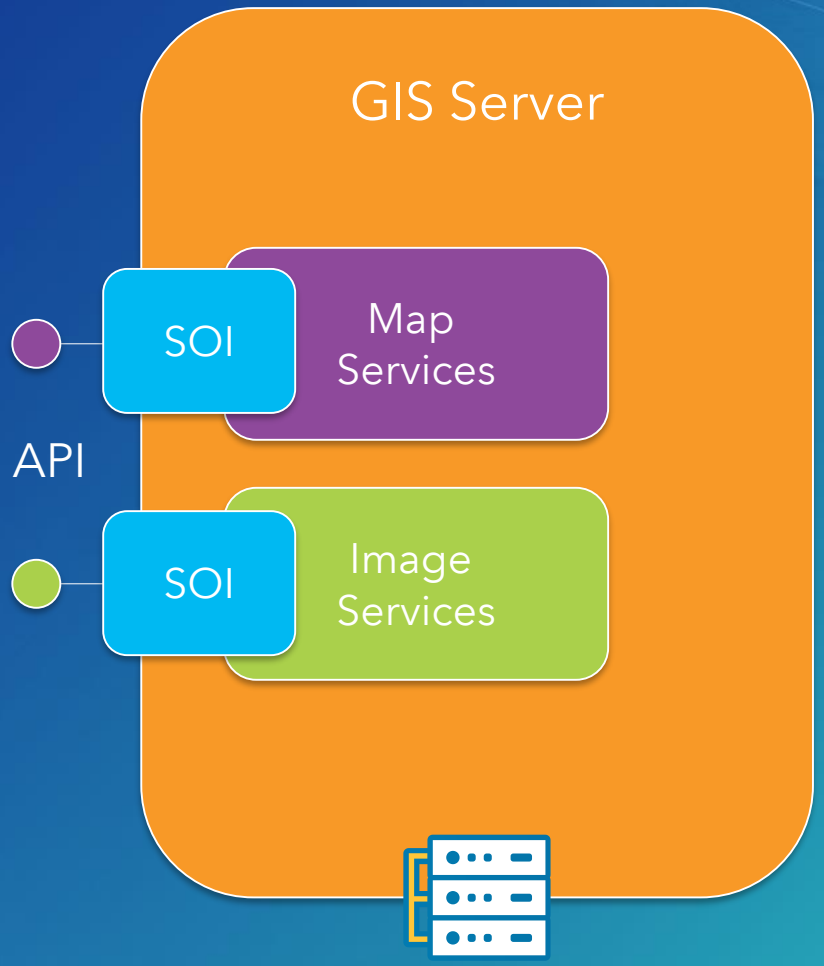
Use cases


Security & Access Control
Data Enrichment
Notifications
Auditing & Metering

Architecture

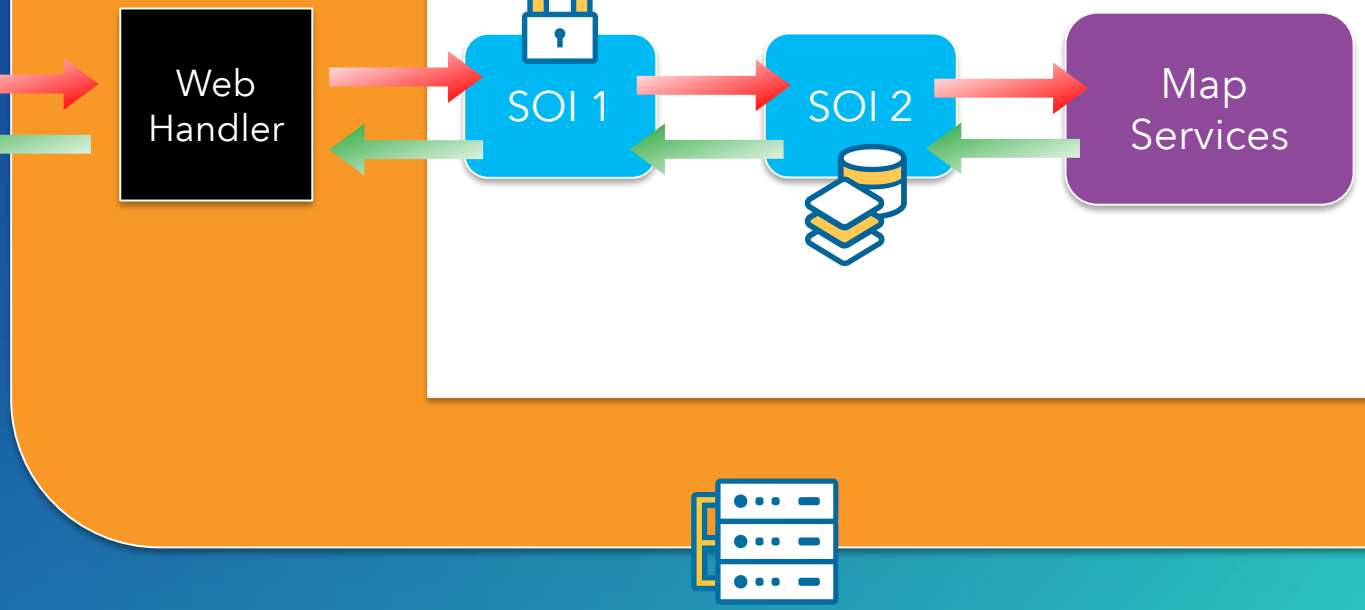
Out-of-box applications

Well-defined API



Inner workings

Out-of-box applications



Demo

Server Object Interceptors

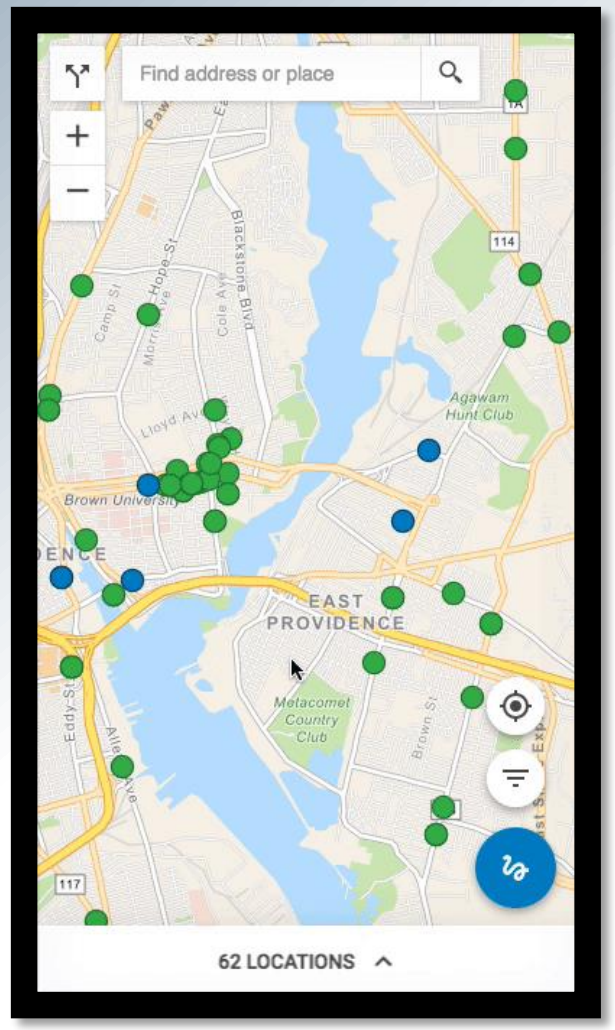
Esri Consulting Services Projects

Carsten Piepel

Custom Application

Mobile sales mapping app

- Responsive app for the mobile sales force
- Workflow:
  1. Find customers and prospects within territory
  2. Show customer info
  3. Create a trip
- JavaScript

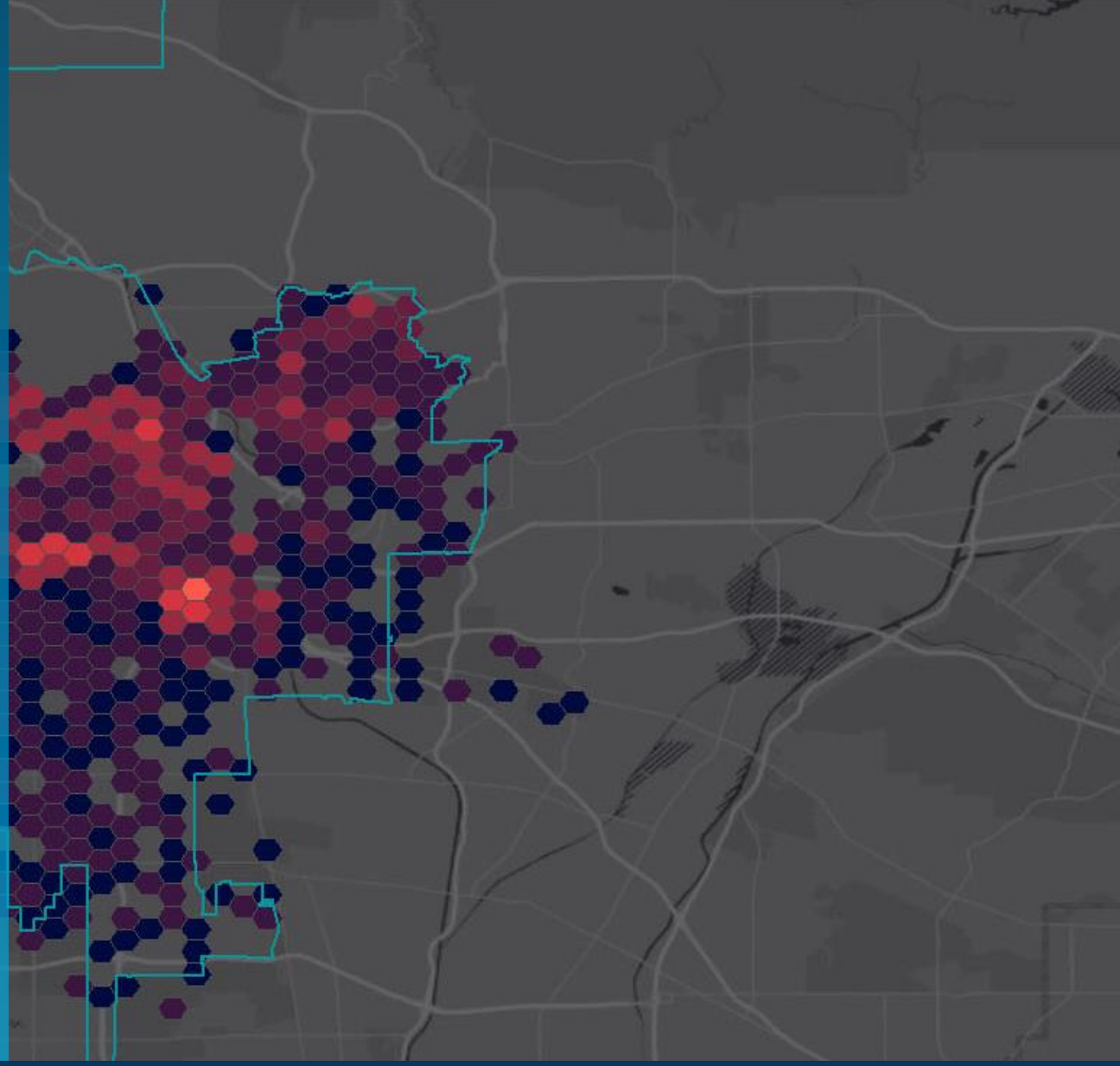


Identify relevant customers and generate route plans

Geoprocessing Service

Hexagon binning

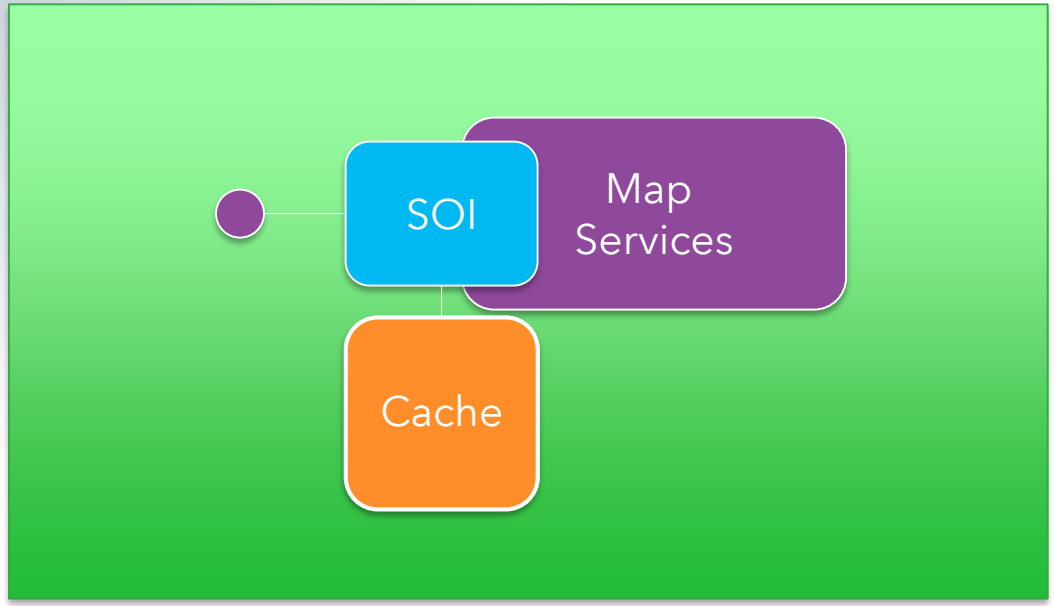
- Aggregate insurance policies into hexagonal bins
- Apply spatial and non-spatial filters
- Asynchronous
- Python and procedural SQL



Highlight concentrations of insurance risk

Caching Aspect

- Part of a bigger security SOI project
- Maintain in-memory cache of ***query*** operation results
- Speeds up frequently repeated queries
- For read-only data
- Java
- Ehcache (Redis, Memached would be better)



Improve query performance by maintaining a cache of frequently accessed query results

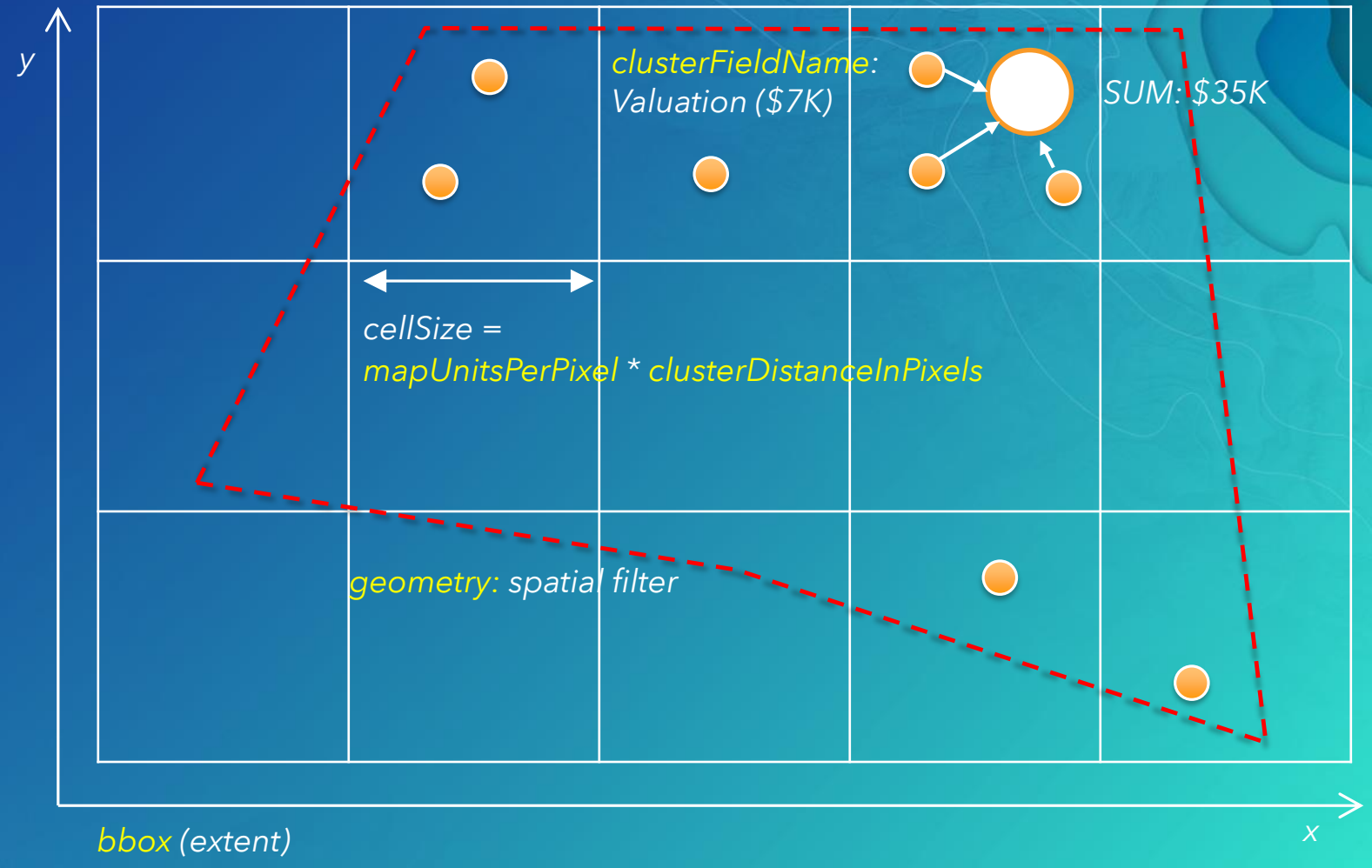
Mapping lots of points

- Problem: Visualize tens of thousands or hundreds of thousands of point features on a web map
- Client-side: Feature layer with out statistics | Cluster layer | Heat map layer | Coming: WebGL-based feature layer
- Server-side: Image layer | Geoprocessing | Clustering

Clustering concepts

1. Query features
2. Assign features to clusters based on regular grid using seed locations
3. Fix clusters:
   - Remove overlap
   - Collapse small neighboring clusters

- Clustering is sensitive to order of features
- Similar to JS API point clustering example
- Thanks to Kerry Coffin



https://github.com/Esri/sever-extension-java

Best Practices & Tips

Samples & IDE integration

Automate development workflows using Admin API

Use GP services for long running tasks

Don’t change the API in an SOI

Implement security carefully

No SOIs for hosted feature or tile requests

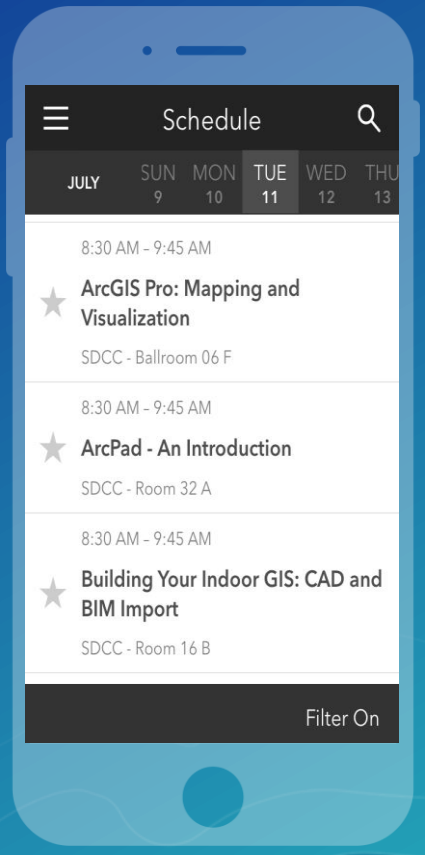
May need to increase ArcSOC heap size

Enable remote debugging for Java from Manager

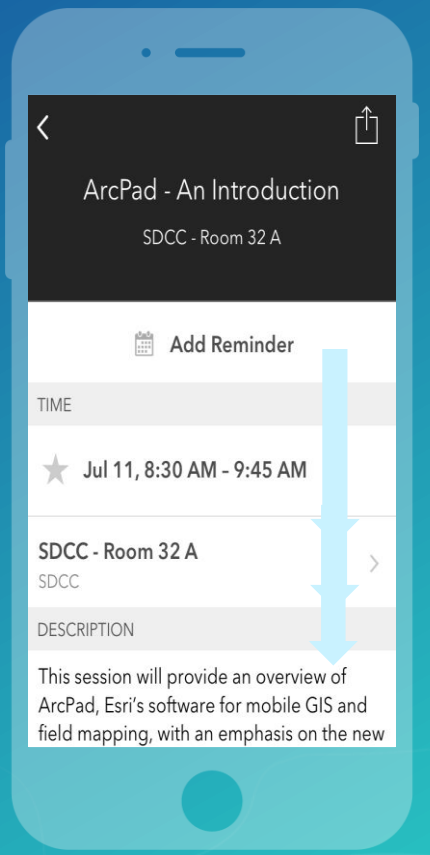
Please Take Our Survey on the **Esri Events App!**

Download the Esri Events app and find your event

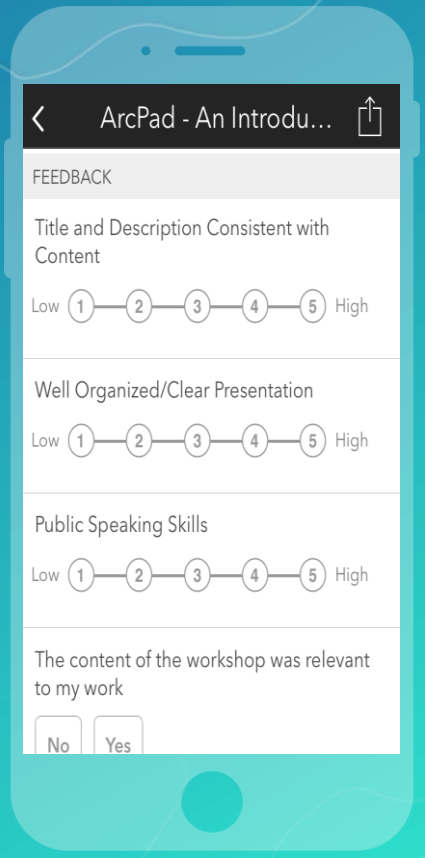
Select the session you attended



Scroll down to find the survey



Complete Answers and Select "Submit"



Questions?

esri
THE
SCIENCE
OF
WHERE