



ArcGIS Runtime SDK for Java: Building Apps

Mark Baird

Jonathan Lavi

2020 ESRI DEVELOPER SUMMIT
Palm Springs

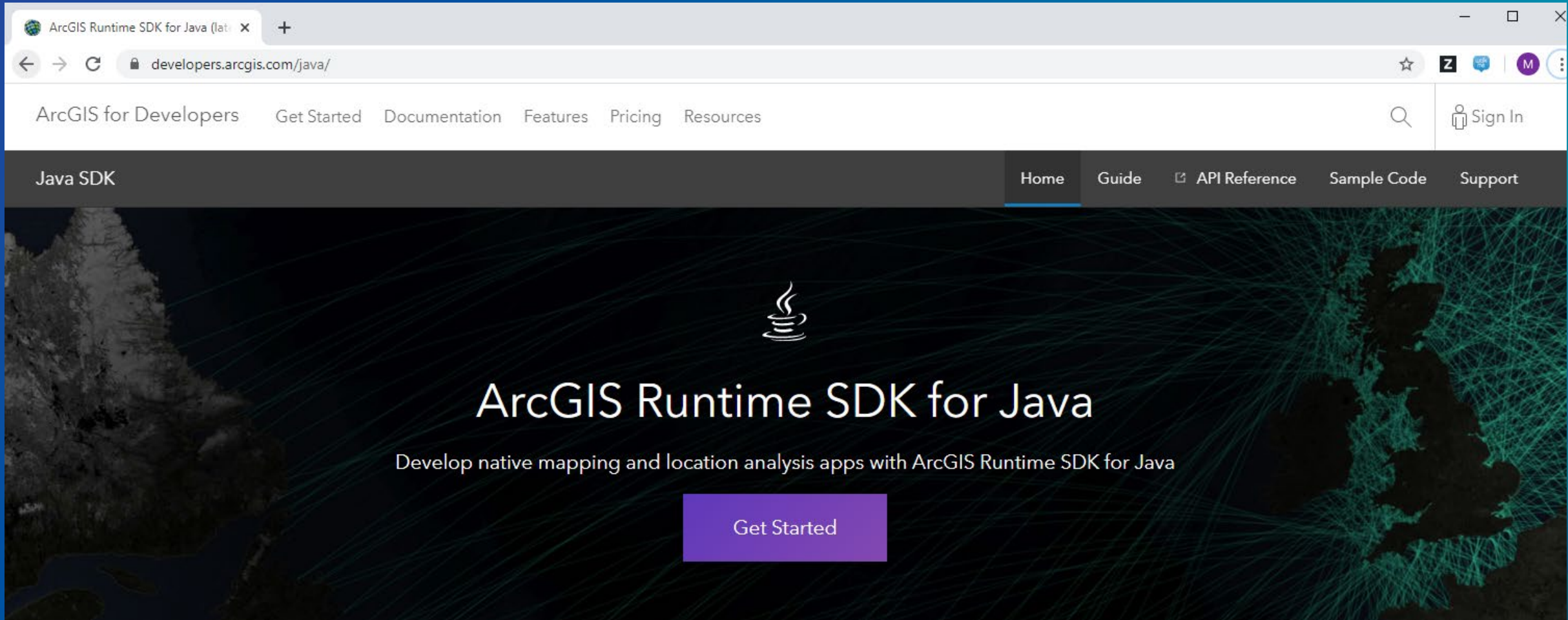
Agenda

- **Getting started with 100.7**
 - **Java 11 / JavaFX**
 - **Base maps, layers and loadable pattern**
 - **Graphics overlays**
 - **Offline maps**
 - **Licensing and deployment**
- 

Getting started with 100.7

An abstract graphic design on a blue background. It features several overlapping, semi-transparent geometric shapes and lines in various colors including green, yellow, pink, and blue. The shapes are layered and oriented diagonally, creating a sense of depth and movement. Some shapes contain patterns like dots or grid lines. The overall composition is dynamic and modern.

Developers website




A screenshot of a web browser displaying the ArcGIS Runtime SDK for Java website. The browser's address bar shows the URL `developers.arcgis.com/java/`. The page features a navigation menu with links for `ArcGIS for Developers`, `Get Started`, `Documentation`, `Features`, `Pricing`, and `Resources`. A search icon and a `Sign In` button are also present. Below the navigation, a dark banner contains the Java logo, the title `ArcGIS Runtime SDK for Java`, the subtitle `Develop native mapping and location analysis apps with ArcGIS Runtime SDK for Java`, and a prominent purple `Get Started` button. The background of the banner is a dark, abstract image with green and blue lines.

ArcGIS Runtime SDK for Java (lat: x +)

← → ↻ 🔒 developers.arcgis.com/java/ ☆ Z M

ArcGIS for Developers Get Started Documentation Features Pricing Resources 🔍 Sign In

Java SDK Home Guide API Reference Sample Code Support



ArcGIS Runtime SDK for Java

Develop native mapping and location analysis apps with ArcGIS Runtime SDK for Java

[Get Started](#)

SDK Installation

- **SDK is available in 2 ways:**
 - Zip / tgz file with all required files
 - Files will be available in a Maven repository
 - Use Gradle scripting
 - Use Maven scripting



Maven™



Java Platform (JDK) 11

Rewind 18 months: What happened to Java?

End of Public Updates for Oracle JDK 8

Oracle will not post further updates of Java SE 8 to its public download sites for commercial use after January 2019. Customers who need continued access to critical bug fixes and security fixes as well as general maintenance for Java SE 8 or previous versions can get long term support through Oracle Java SE Subscription or Oracle Java SE Desktop Subscription. For more information, and details on how to receive longer term support for Oracle JDK 8, please see the [Oracle Java SE Support Roadmap](#).

Oracle to charge for Java from Jan 2019

May 1st, 2018 · 13 Comments

Oracle have announced that, "after January 2019", Java SE 8 public updates will not be available for "business, commercial or production use" without a commercial license.

Organisations will now need to take stock of all their software running Java SE 8 and start to work out what potential bill they are looking at next year.

Finding Java

The Java package – that will already be available within your organisation – includes a tool called "Java Usage Tracker" that will report on:



JavaFX will be removed from the Java JDK

Oracle's rich media web technology will be broken out of standard Java as of JDK 11, and will proceed at its own pace

Java 11 is here, Java 14 is coming!

- 100.4 works with Java 8 and 11
- 100.5+ is for Java 11 only
- 100.8 is Java 11 and 14
- You can choose your JRE
 - AdoptOpenJDK
 - OpenJDK
 - Oracle JDK
- Java FX is no longer in the JDK
- Separate binaries are available



Maven[™]

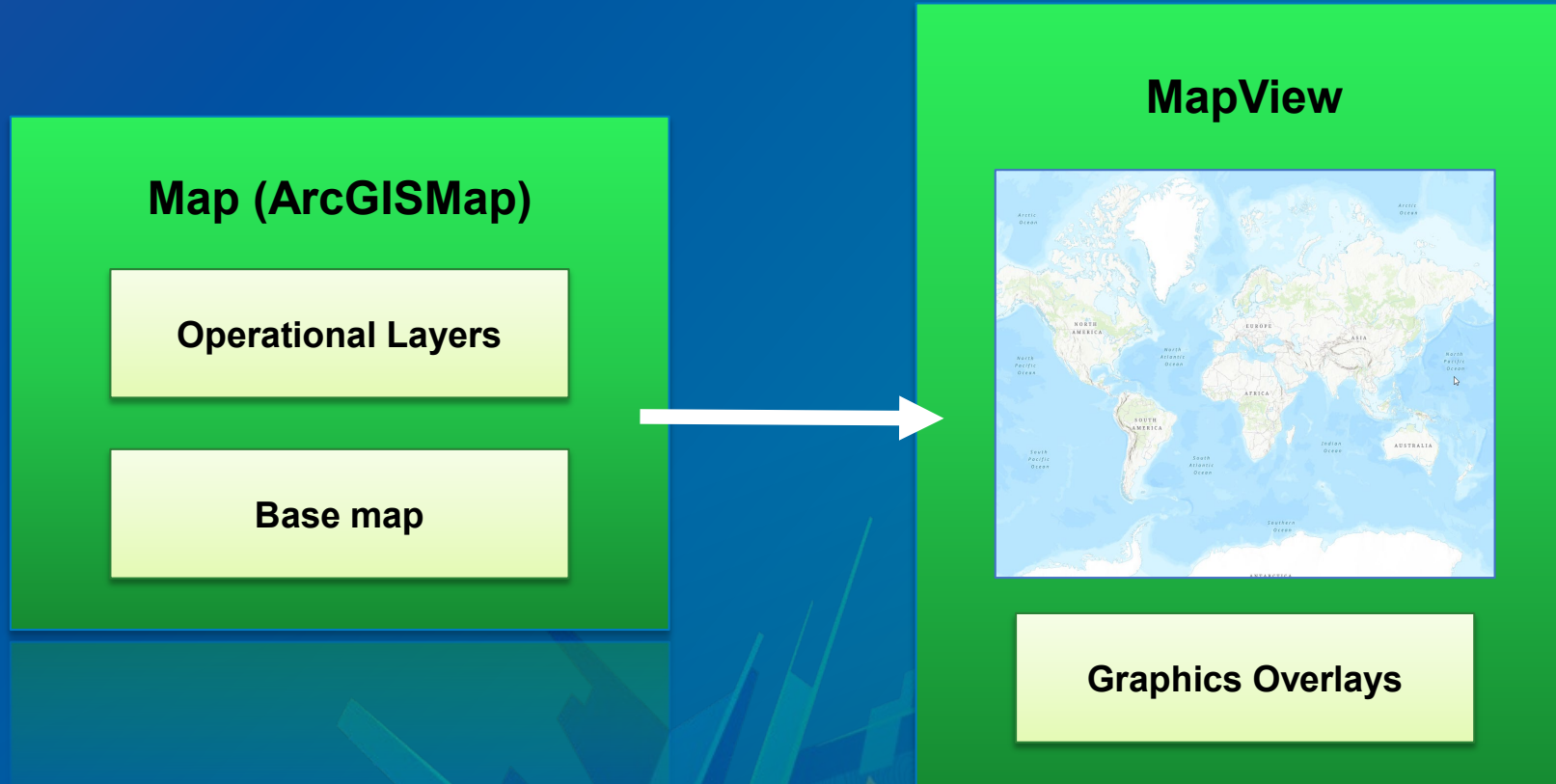


- <https://developers.arcgis.com/java/latest/guide/end-of-public-updates-for-oracle-s-java-se-8.htm>

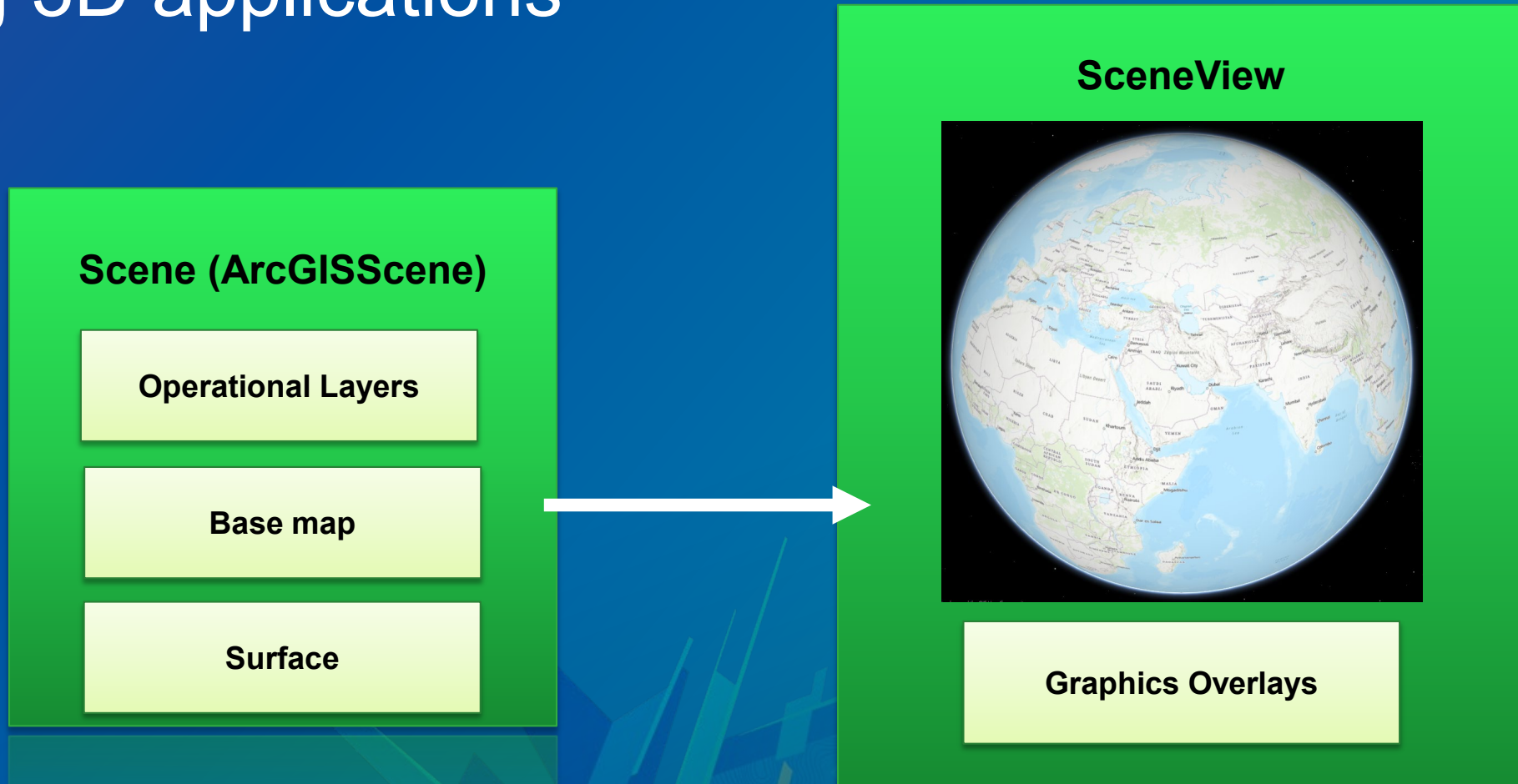
The background features a vibrant blue gradient. On the right side, there is a complex, layered graphic composed of various geometric shapes, including triangles and rectangles, in shades of green, yellow, pink, and blue. Some of these shapes contain patterns like dots or lines. In the top-left corner, there are several overlapping, semi-transparent rectangular shapes in shades of green, yellow, and pink. The overall aesthetic is modern and digital.

Base maps, layers and loadable pattern

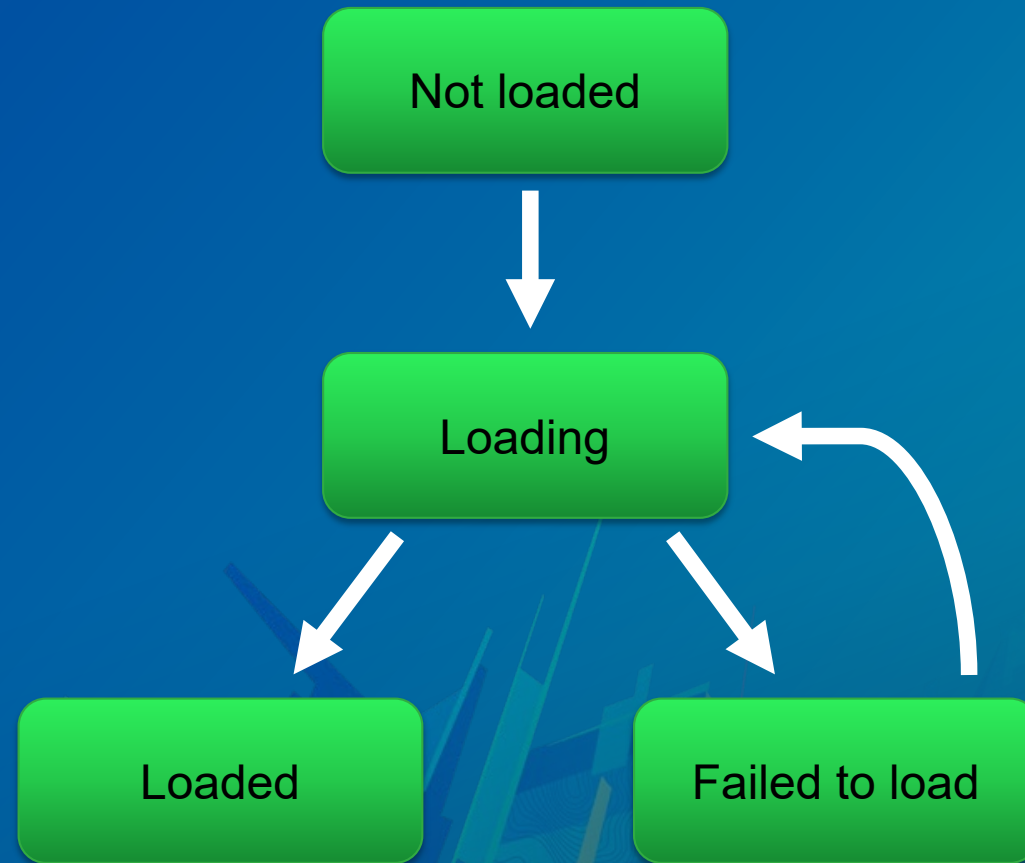
Writing 2D applications



Writing 3D applications



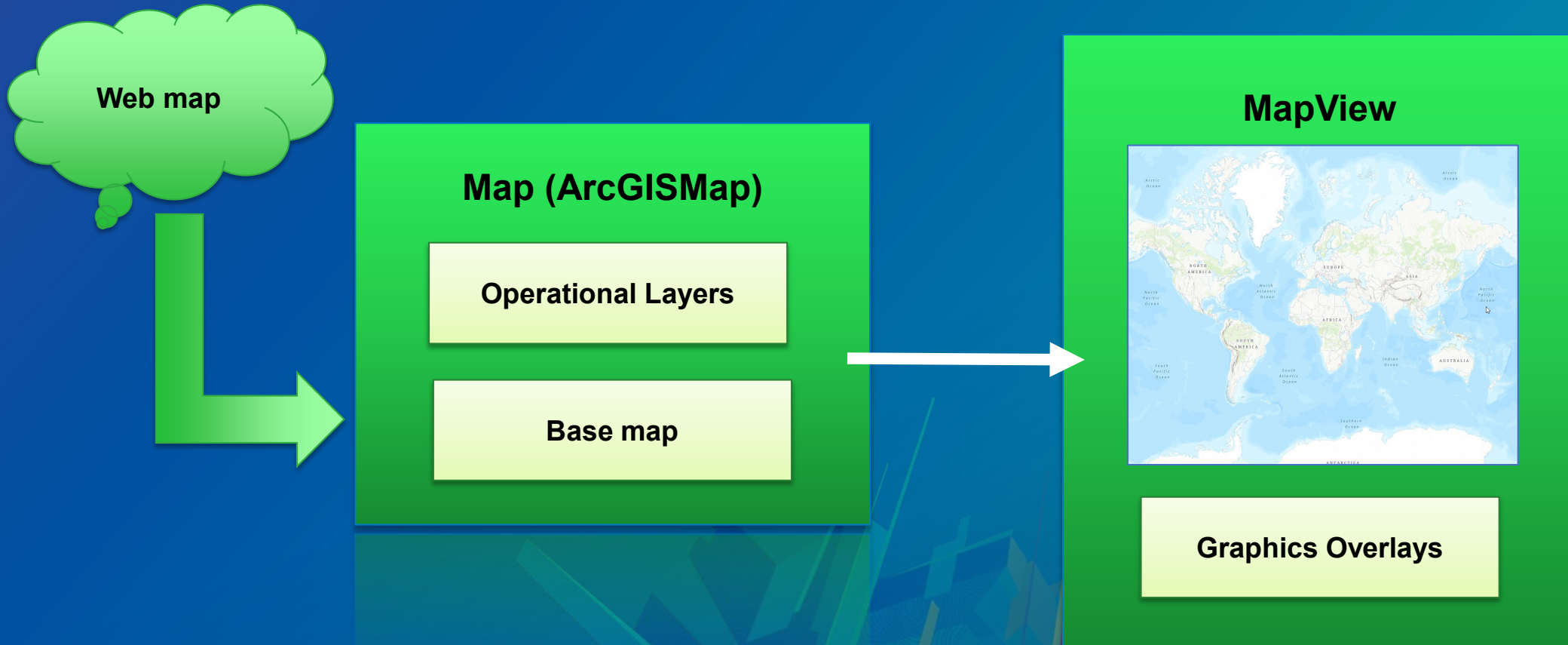
Loadable Pattern





Demo: Maps and layers

Writing 2D applications

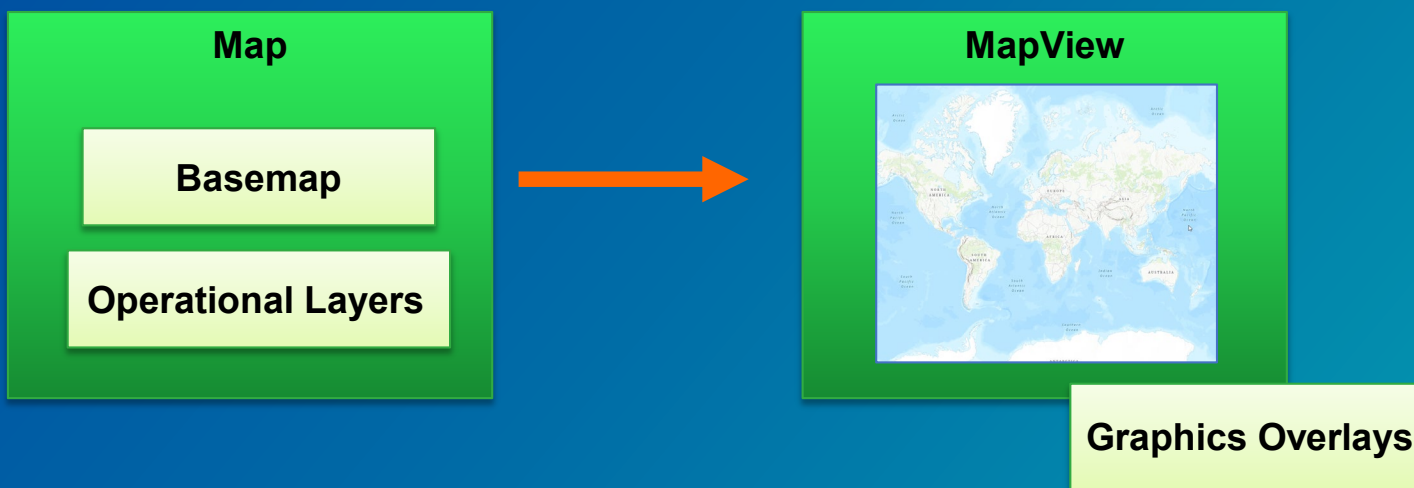
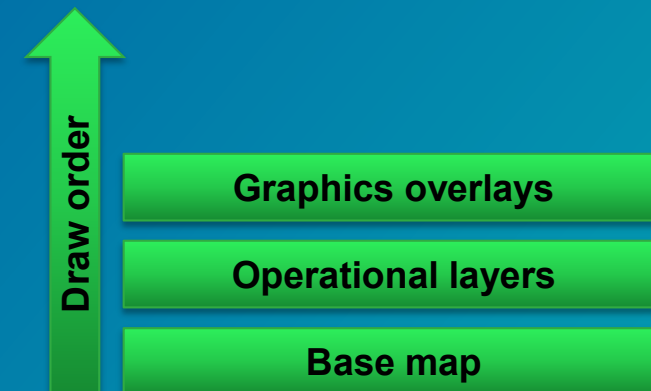


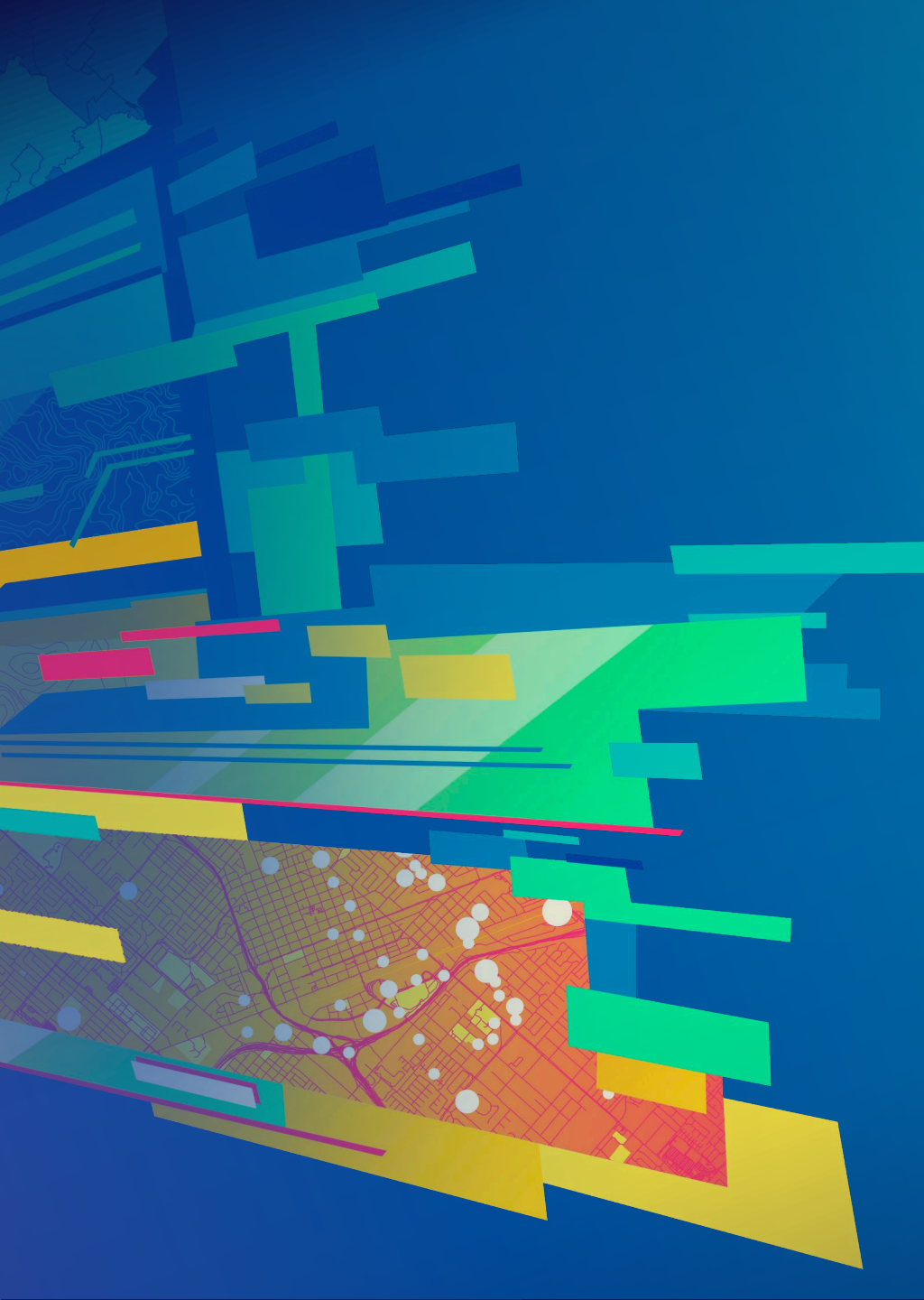
Graphics Overlays



Graphics overlays

- Used to display temporary or fast updating items
- Can be used to display point, lines or polygons
- Can be rendered in the same way as feature layers
 - Symbols, Unique value renderers, etc
- Graphics overlays are part of the MapView or SceneView
- Not persisted





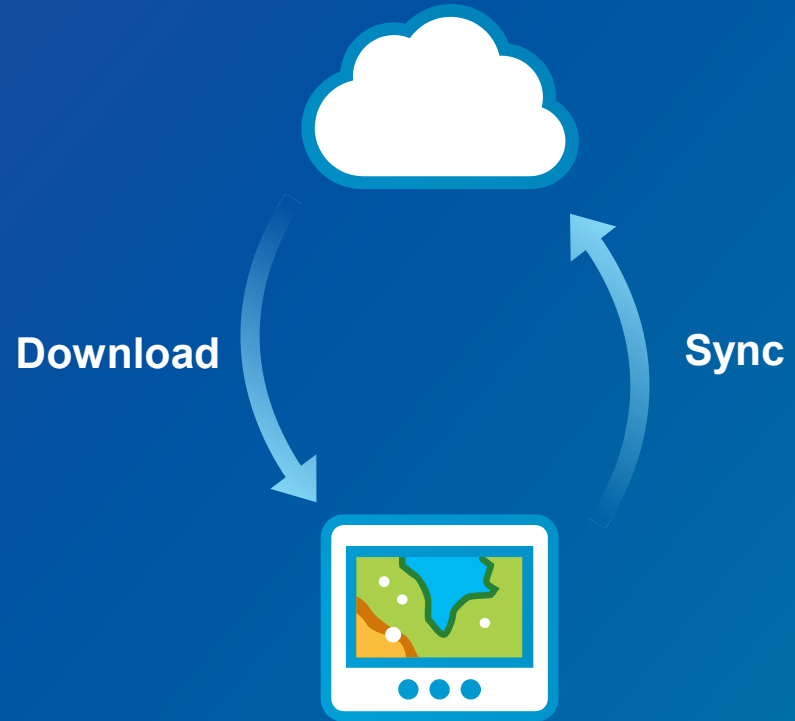
Demo: Graphics Overlays

Offline Data



Offline

Services Pattern



Desktop Pattern



The background features a vibrant blue gradient. On the right side, there is a complex, layered graphic design. It includes several overlapping, semi-transparent geometric shapes in shades of green, yellow, and pink. A map of Europe is visible, rendered in a light blue color, partially obscured by the other elements. The overall aesthetic is modern and digital.

Licensing and deployment

License your application

- **Choose a license level:**
 - Lite
 - Basic
 - Standard
 - Advanced
- See <https://developers.arcgis.com/java/latest/guide/license-your-app.htm>
- **Choose license method**
 - Named user
 - License key

License key

- Use the `setLicense` method before your app uses ArcGIS functionality

```
public static void main(String[] args) {  
    // set license  
    ArcGISRuntimeEnvironment.setLicense("runtimelite,1000,ru  
  
    // launch application  
    Application.launch(args);  
}
```

Named user licensing

- Log into your portal to get the license key

```
private void licenceAppOnline() {
    // connect to ArcGIS Online or a portal with a named user
    // Note this is very insecure code and you should consider using
    // something like oAuth instead for example!
    {
        UserCredential credential =
            new UserCredential( username: "username", password: "password" );

        // connect to the portal with my credential
        portal = new Portal("https://your-org.maps.arcgis.com/");
        portal.setCredential(credential);

        // load the portal
        portal.loadAsync();
    }
}
```

Named user licensing

- Log into your portal to get the license key

```
private void licenceAppOnline() {
    // connect to ArcGIS Online or a portal with a named user
    // Note this is very insecure code and you should consider using
    // something like oAuth instead for example!
    UserCredential credential =
        new UserCredential( username: "username", password: "password" );

    // connect to the portal with my credential
    portal = new Portal("https://your-org.maps.arcgis.com/");
    portal.setCredential(credential);

    // load the portal
    portal.loadAsync();
}
```


Named user licensing

- Log into your portal to get the license key

```
private void licenceAppOnline() {  
    // connect to ArcGIS Online or a portal with a named user  
    // Note this is very insecure code and you should consider using  
    // something like oAuth instead for example!  
    UserCredential credential =  
        new UserCredential( username: "username", password: "password" );  
  
    // connect to the portal with my credential  
    portal = new Portal("https://your-org.maps.arcgis.com/");  
    portal.setCredential(credential);  
  
    // load the portal  
    portal.loadAsync();  
}
```

Named user licensing (part 2)

```
// listen for the portal loaded event on a new thread
portal.addDoneLoadingListener(() -> {
    // check the load status was successful
    if (portal.getLoadStatus() == LoadStatus.LOADED) {

        // get the license info from the server
        ListenableFuture<LicenseInfo> licenceInfoFuture = portal.fetchLicenseInfoAsync();

        licenceInfoFuture.addDoneListener(() -> {
            try {
                // get the license info
                licenseInfo = licenceInfoFuture.get();

                // apply the license to the app
                ArcGISRuntimeEnvironment.setLicense(licenseInfo);

                // save for offline use later
                savedLicenseJSON = licenseInfo.toJson();
            }
        });
    }
});
```

Named user licensing (part 2)

```
// listen for the portal loaded event on a new thread
portal.addDoneLoadingListener(() -> {
    // check the load status was successful
    if (portal.getLoadStatus() == LoadStatus.LOADED) {

        // get the license info from the server
        ListenableFuture<LicenseInfo> licenceInfoFuture = portal.fetchLicenseInfoAsync();

        licenceInfoFuture.addDoneListener(() -> {
            try {
                // get the license info
                licenseInfo = licenceInfoFuture.get();

                // apply the license to the app
                ArcGISRuntimeEnvironment.setLicense(licenseInfo);

                // save for offline use later
                savedLicenseJSON = licenseInfo.toJson();
            }
        });
    }
});
```

Named user licensing (part 2)

```
// listen for the portal loaded event on a new thread
portal.addDoneLoadingListener(() -> {
    // check the load status was successful
    if (portal.getLoadStatus() == LoadStatus.LOADED) {

        // get the license info from the server
        ListenableFuture<LicenseInfo> licenceInfoFuture = portal.fetchLicenseInfoAsync();

        licenceInfoFuture.addDoneListener(() -> {
            try {
                // get the license info
                licenseInfo = licenceInfoFuture.get();

                // apply the license to the app
                ArcGISRuntimeEnvironment.setLicense(licenseInfo);

                // save for offline use later
                savedLicenseJSON = licenseInfo.toJson();
            }
        });
    }
});
```

Named user licensing (part 2)


```
// listen for the portal loaded event on a new thread
portal.addDoneLoadingListener(() -> {
    // check the load status was successful
    if (portal.getLoadStatus() == LoadStatus.LOADED) {

        // get the license info from the server
        ListenableFuture<LicenseInfo> licenceInfoFuture = portal.fetchLicenseInfoAsync();

        licenceInfoFuture.addDoneListener(() -> {
            try {
                // get the license info
                licenseInfo = licenceInfoFuture.get();

                // apply the license to the app
                ArcGISRuntimeEnvironment.setLicense(licenseInfo);

                // save for offline use later
                savedLicenseJSON = licenseInfo.toJson();
            } catch (Exception e) {
                // handle exception
            }
        });
    }
});
```



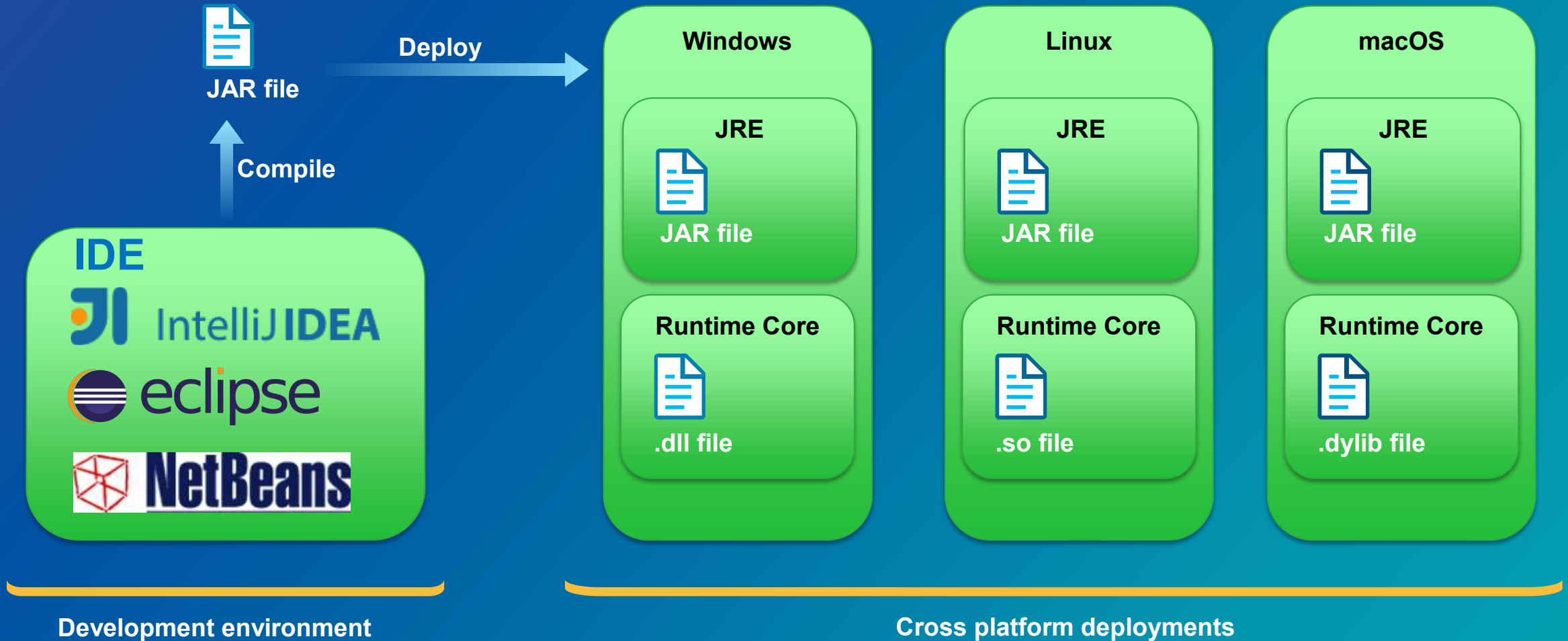
Named user licensing (Offline)

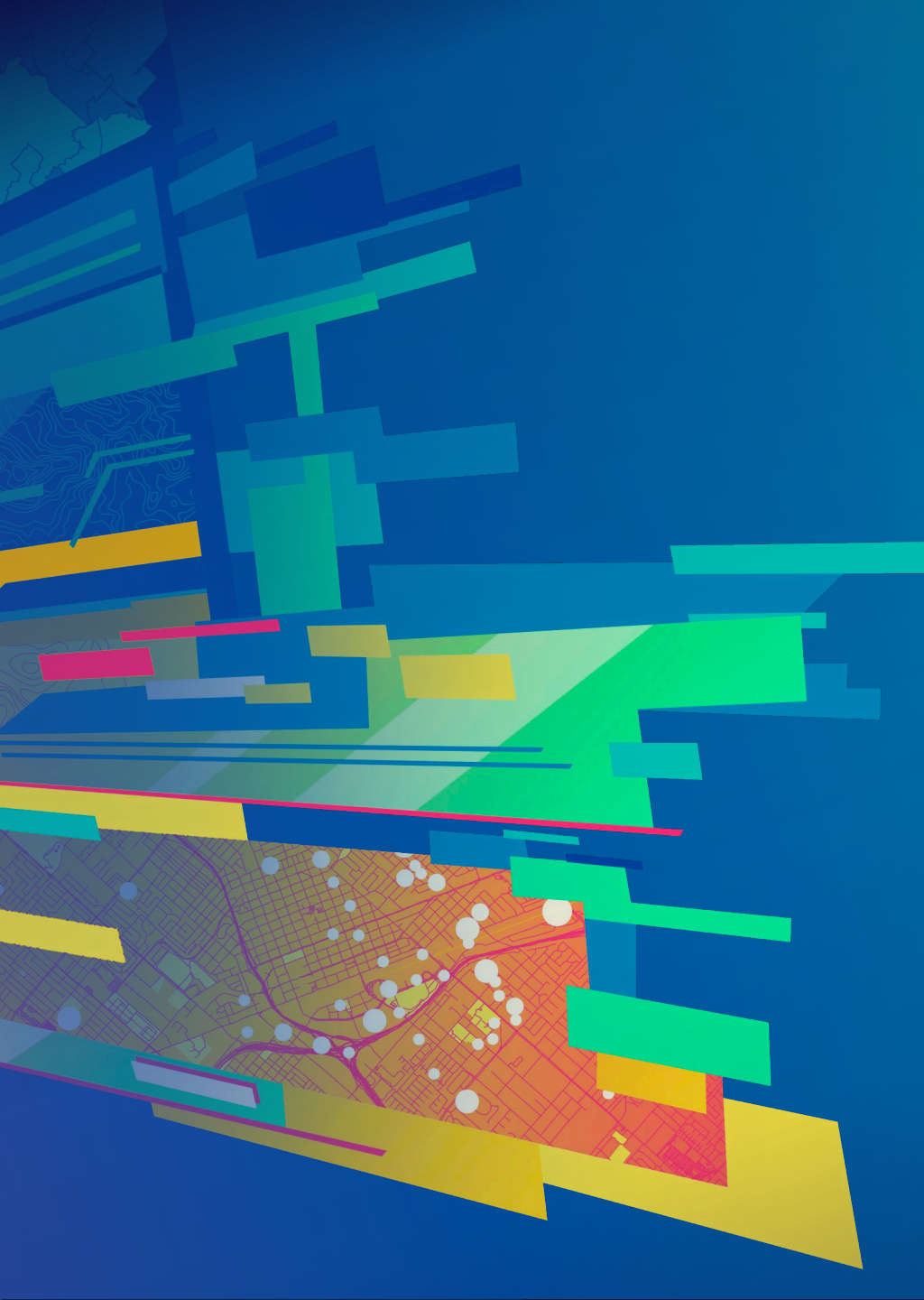


```
// create a licenseInfo from the saved license JSON
LicenseInfo licenseInfo = LicenseInfo.fromJson(savedLicenseJSON);

// apply the license to the app
ArcGISRuntimeEnvironment.setLicense(licenseInfo);
```


Java Runtime development and deployment





Demo: Deployment

Summary

- **Cross platform Java 11 / Java FX desktop apps**
 - **Key classes and techniques for mapping apps**
 - **Graphics overlays**
 - **Offline maps**
 - **License and deployment**
- 



esri

THE
SCIENCE
OF
WHERE