Zebra DevCon 2023

ZEBRA

# Spatial Computing = AR + ML
# Your New Super Power

**Dave Koch**
EMC Director Emerging Technology

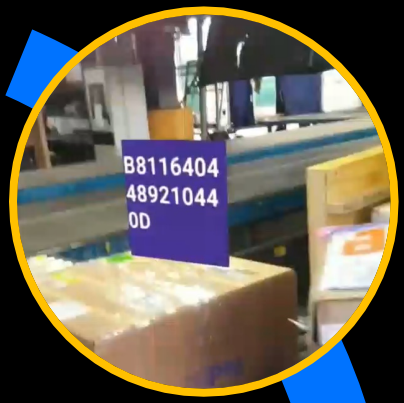
# Spatial Computing = ML + AR

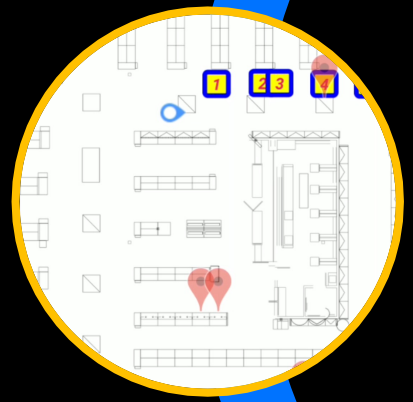
## Recent innovations enable breakthrough applications

Zebra DevCon 2023

- 24 Products in Zebra's EMC portfolio are ARCore certified
- Spatial Computing makes the mobile device spatially aware and every function it performs
- Zebra's breakthrough retail product recognition stack
- Leveraging high-performance Qualcomm AI accelerators on chip

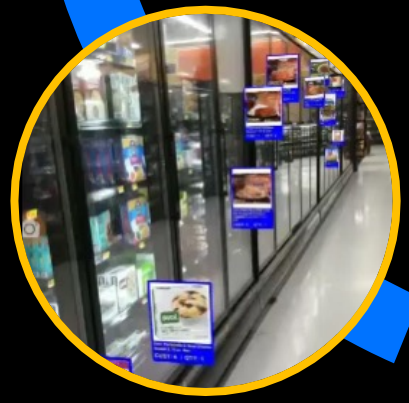
Zebra DevCon 2023

Enterprise Use Cases are limited

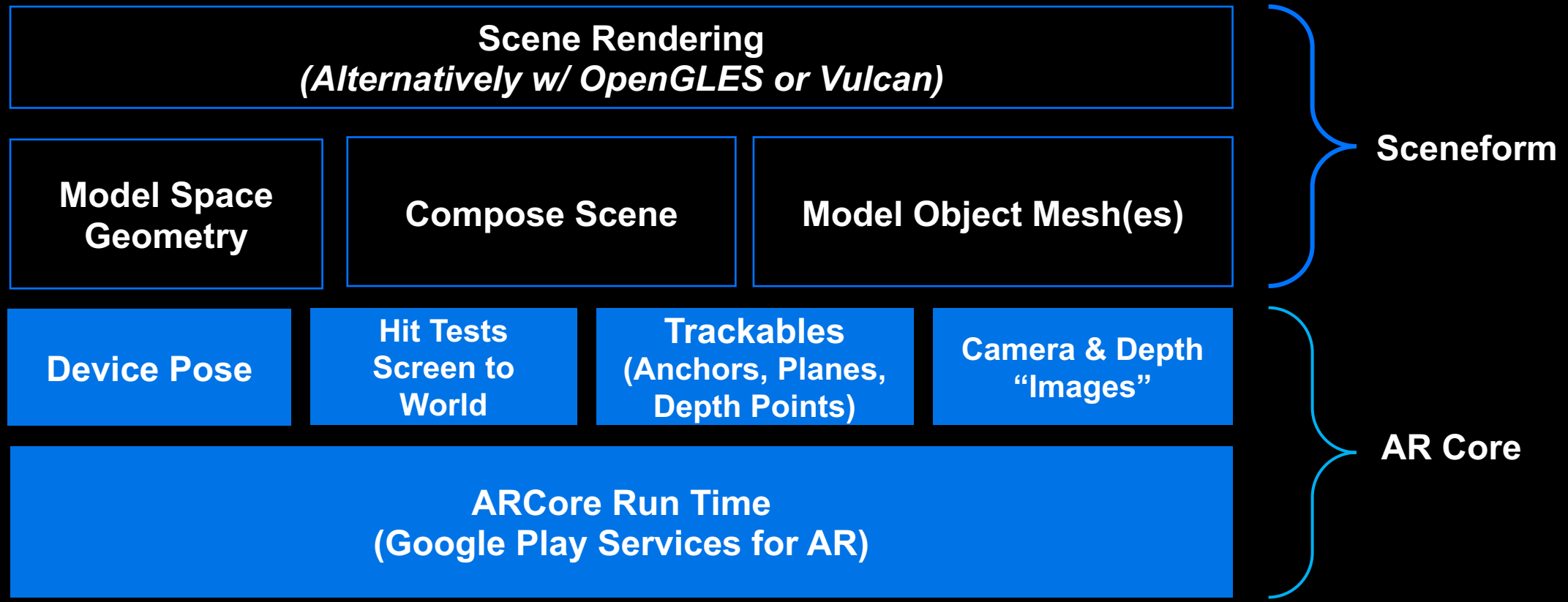
**Only by our imagination**

# Your App on top of ARCore

## API Overview

Zebra DevCon 2023



# Get Started w/ AR Banner Placement

## Sample Code

Hello DevCon2023!

Zebra DevCon 2023

### Banner Placement w/ User Tap

```
override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
  arSceneView = binding.sceneformArSceneView
  arSceneView.setOnTouchListener { _, event ->
   if (event.action == MotionEvent.ACTION_UP) {
      arSceneView.arFrame?.hitTest(event)
         ?.minByOrNull { it.distance }
         ?.addNode(scene)
         ?.addBanner(context, text = "Hello \n DevCon2023", fontSize = 20)   }}

fun HitResult.addNode(scene: Scene): Node {
   val node = Node()
   node.worldPosition = createAnchor().position()
   val plane = trackable as Plane
   node.setLookDirection(plane.centerPose.yAxis.toVector3().negated())
   return node }

fun Node.addBanner(context: Context, image: Bitmap?, text: String?, fontSize: Int){
   ViewRenderable.builder()
      .setView(context, R.layout.banner_viewrenderable_layout).build()
      .thenAccept { renderable ->
            renderable.view.findViewById<TextView>(R.id.banner_line_1).apply {
               setText(text)
               textSize = fontSize.toFloat()}
               image?.apply {
                 renderable.view.findViewById<ImageView>(R.id.odp_image).apply {
                   setImageBitmap(image)  }}}
         this.renderable = renderable  }}
```

### Banner Placement w/ Scanner

```
fun createDataWedgeProfile(context: Context, barcodeReceiver: BroadcastReceiver) {
   val configBundle = Bundle()
   val bConfig = Bundle()
   val bParams = Bundle()
   val bundleApp1 = Bundle()
   val appName = context.packageName
   bParams.putString("scanner_selection", "auto")
…..
   filter.addAction(Activity_Intent_Filter)
   filter.addAction(NOTIFICATION_ACTION)
   context.registerReceiver(barcodeReceiver, filter)

 private val barcodeReceiver = object : BroadcastReceiver() {
       override fun onReceive(context: Context?, intent: Intent?) {
          val action = intent?.action
          if (action == Activity_Intent_Filter) {
             val decoded = Intent.getStringExtra(Intent_Key_Data) ?: "none"
             sharedViewModel.setBarcode(decoded)
          }}}
```

```
val scannerDirection = camera.up //scanner 90 degrees rotated from camera
val scannerPosition = camera.worldPosition.add(Vector3(-0.034f, 0f, 0f)) //scanner offset
val scannerRay = Ray(scannerPosition, scannerDirection)

scannerRay.hitResult(arSceneView) //hit test against the scene vert plane
          ?.minByOrNull { it.distance }
          ?.addNode(scene)
          ?.addBanner(context, text = $decoded, fontSize = 20)
```

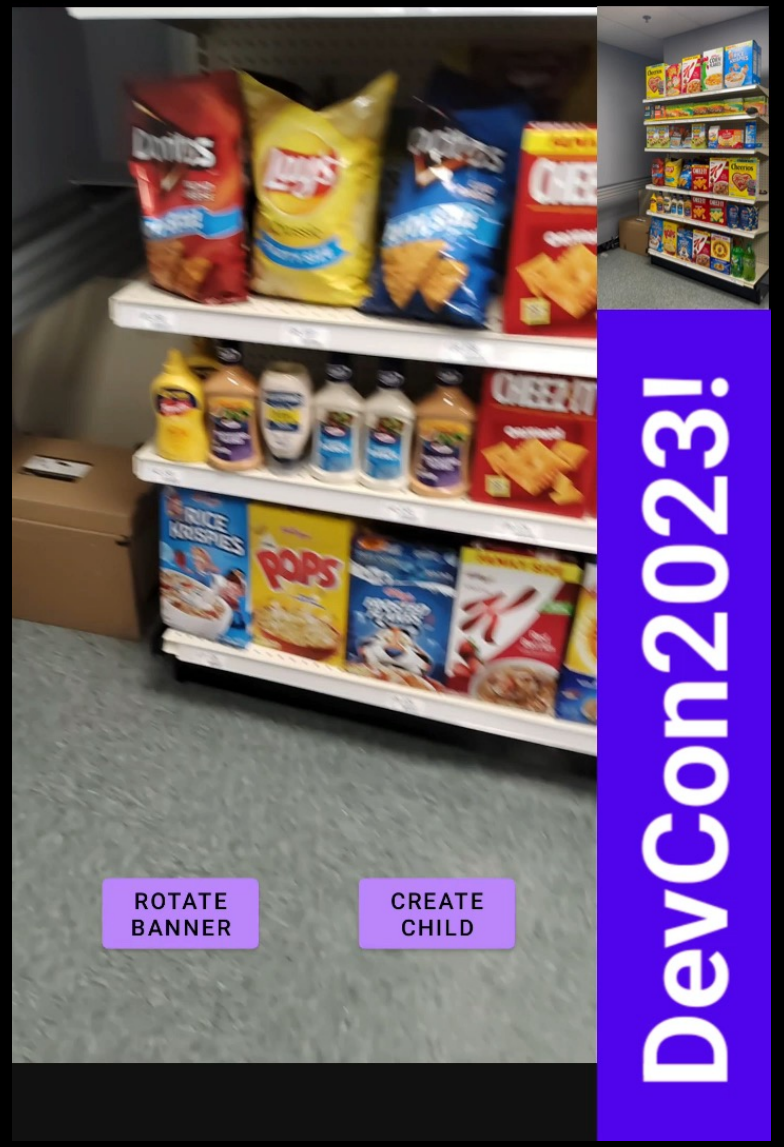
# Let run the sample app!

## Hello DevCon2023

- A simple app that places banners on vertical surfaces

- Illustrates
  - Configuring ARCore and Sceneform
  - Create Banner Renderable
  - Create a Child Banner
  - Rotation of Banner
  - Raycast barcode scans, visualized as Banners

Sample code will be made available upon request

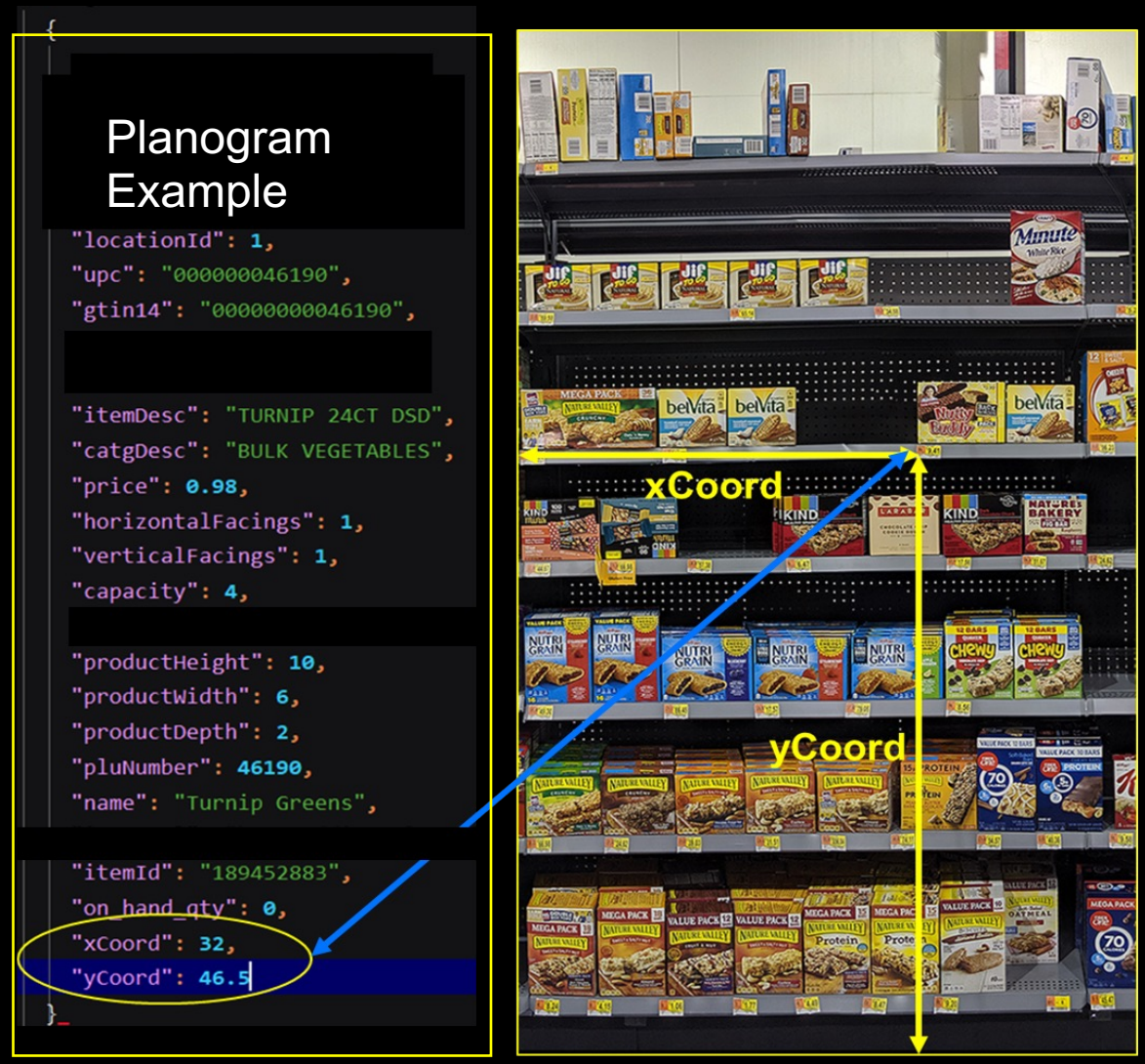


# Retail Use Cases

## To level set…

- Planogram
  - Placement of shelf labels by UPC, Name, Price, section, shelf, XY location, vertical and horizontal facings, slot capacity, product width, height, depth…
- Realogram
  - Placement of the shelf labels on each section AS placed
- Planogram Compliance
  - Are Shelf Labels placed correctly per Planogram schematic
  - Are Products are placed correctly corresponding to the Realogram?
- Shelf Health
  - Out of Stock
  - Share of Shelf (for CPG)
  - Inventory Level (vs slot capacity)
- Online Order Picking
  - Support Picking apps
  - Indoor nav/wayfinding

Zebra DevCon 2023



# Planogram – Realogram – Compliance – Shelf Health – Order Pick

## Enabled by On Device Product ID + Spatial Computing

Zebra DevCon 2023

Planogram
Shelf Label UPC
Position on Shelf
Facings

Defined Centrally & Pushed Down to individual Stores

Realogram
Shelf Label UPC
Position on Shelf
Facings

As implemented physically in each store

Product Placement on Shelf

Snapshot of current state of products on shelves

Planogram Compliance

Implementation Errors

Order Picking

Shelf Health

Status of “slot”

- Gaps
- Facings vertical and horizontal
- Spreads
- Out of Stock
- Low Inventory*

```
{Store:  4480
Time: 012/02/2022 12:30PM
Aisle: 4
Section: 12
Products:

 UPC: 041789007019
 Product: Maruchan Yakisoba Japanese Chicken Noodles 4 Oz.
 Stock Level: Medium
 PGCompliance: Good
 PlacementCompliance: “Facing Gap”
 Shelf: 3
 SlotOrder: 5

...
}
```

# Realogram Generation

## Why and How

Zebra DevCon 2023

- Why?
  - Verifying compliance against the Planogram Schematic
  - Replaces Planogram for Retailers who don’t deploy top-down planogram schematic for each store
  - Feeds store picking solutions to guide pickers to the products on shelf.
- How Done Today?
  - Manual scanning of barcodes with an app
    - Enter section and shelf number
    - Scan each barcode in sequence
    - Time consuming and error prone
    - Does NOT provide true XY coordinates – just the shelf number and sequence
    - Does NOT verify facings (horizontal and vertical)
    - Only done once per store on setup
  - Many retailers just assume correct label placement per planogram schematic
    - Unless discovered otherwise while picking or stocking

# Multiple Shelf Label Recognition for Realogram

## Shelf Label Realogram

# On Device Product Recognition

# Realtime Shelf Health

# On Line Order Picking in Store

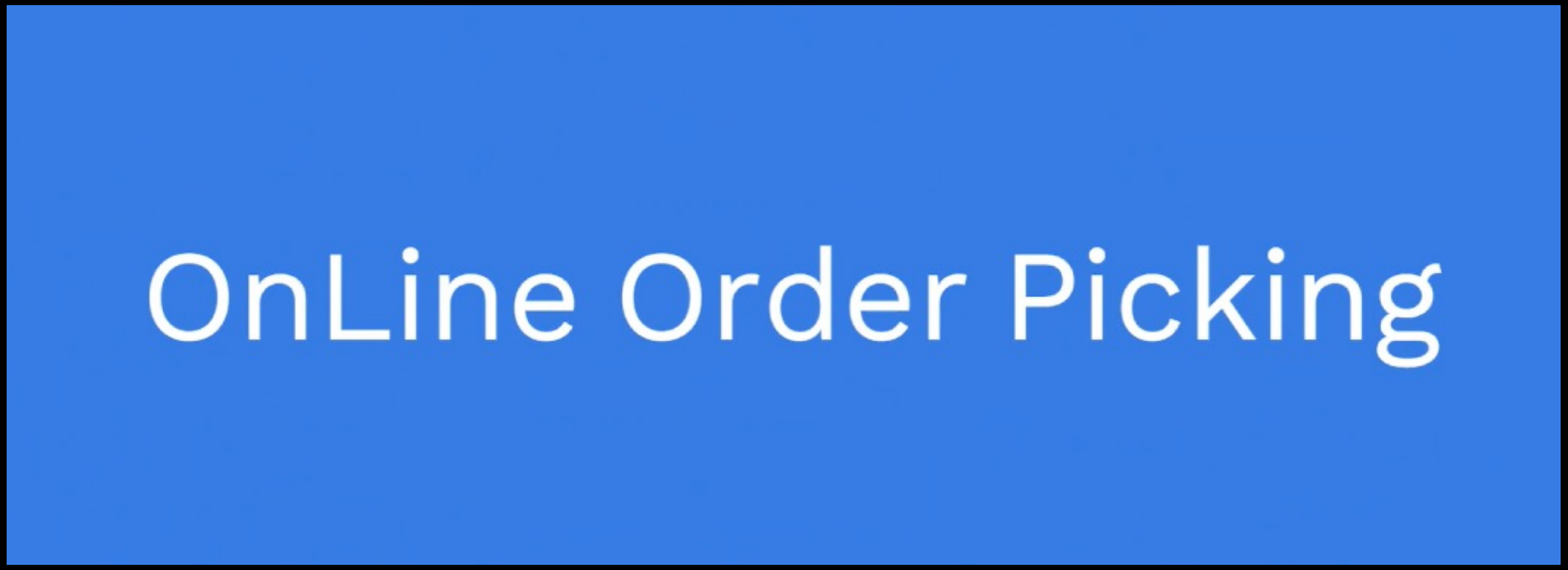
## Putting it all together!

Zebra DevCon 2023

- Start with Planogram (if available)
- Create Instant Realogram (1 minute per section)
- Compare Planogram with Realogram for compliance
- Periodic cycle counting – update Realogram
- Current inventory aware database of every shelf label “content”
- Feed any location aware picking app with the up-to-date location
- Like so…

# Order Picking and Store Navigation

## Online Order Picking



# Questions

Zebra DevCon 2023

ZEBRA

# Thank You