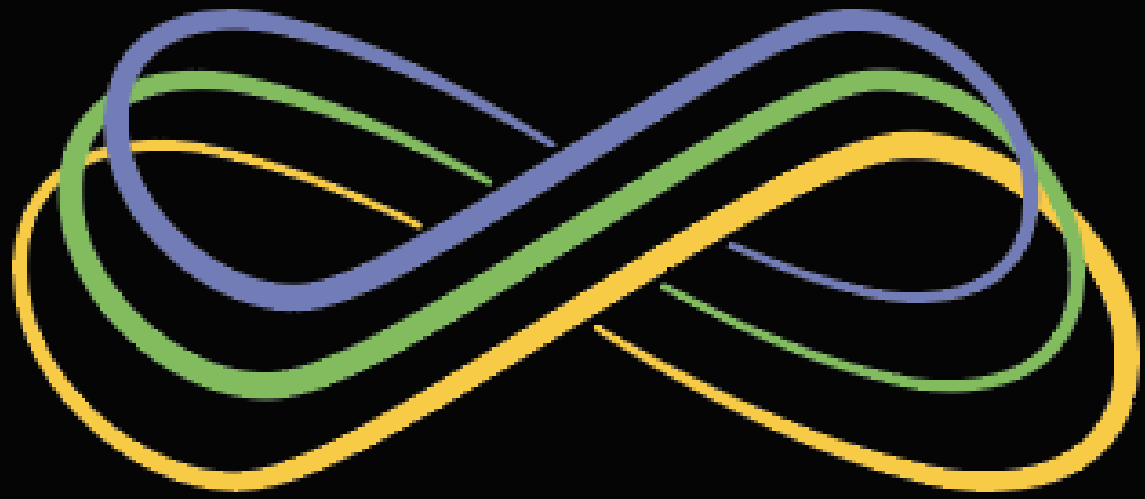
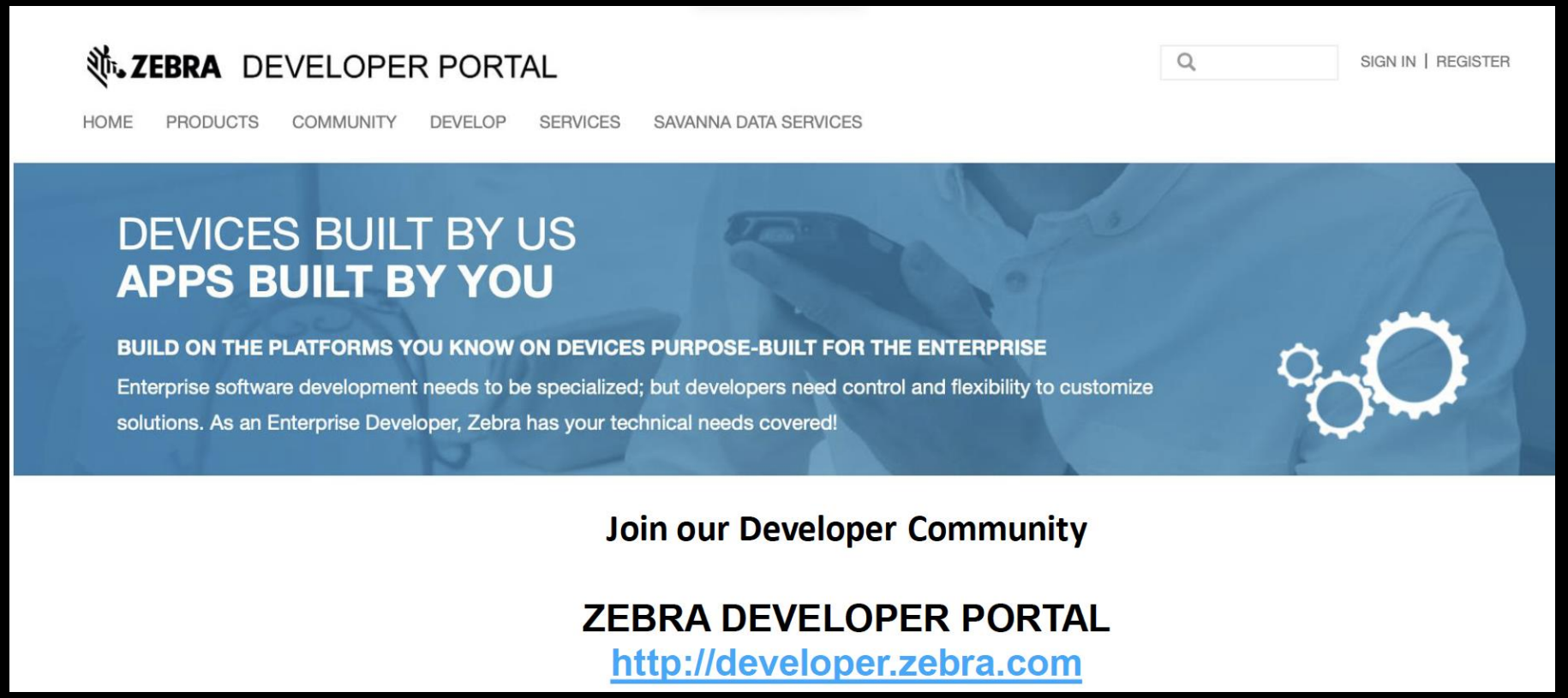


Zebra Developers

Build Your Edge

Zebra
Developers
Build Your Edge



Join our Developer Community

ZEBRA DEVELOPER PORTAL
http://developer.zebra.com

Zebra Developers
Build Your Edge

Zebra DevTalk - Optimizing the Link-OS Multiplatform SDKs

Manuel Caicedo-Rivera

Senior Software Engineer

ISV Validation & Integration Lead

manuelcaicedo

- APIs & SDKs
- Why Optimization
- LINK-OS SDK Case Study
- Q&A

SDKs & APIs

Why are SDKs & APIs the new treasure?

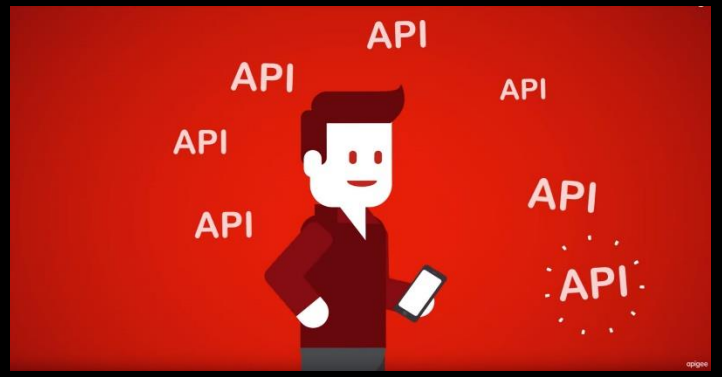
- *APIs can reduce operational or technology costs by simplifying and accelerating development*
- *APIs greatly simplified the integration process*
- *API market gaining momentum*
- *APIs allow business to monetize data*

McKinsey Study

Video: “What's an API?”

Set of protocols and defined methods between several software components

- Software interfaces that let different computer systems and applications talk to one another
- Gives developers frictionless and secure access to company data so they can build applications easily
- Digital building blocks



Multiplatform SDK

Improving the Developer Experience

SDK

- *APIs*
- *Software Package*
- *Software Framework*
- *Best Practices*
- *Development Platforms*
- *IDEs Plugins*

DEVELOPMENT TOOLS

Experience the faster and smarter way to easily integrate printers with the devices and applications your customers use today.

- Multiplatform SDK
- Cloud Connect
- Browser Print
- Print Connect
- Network Connect
- Card Studio 2.0

Powered by Link-OS

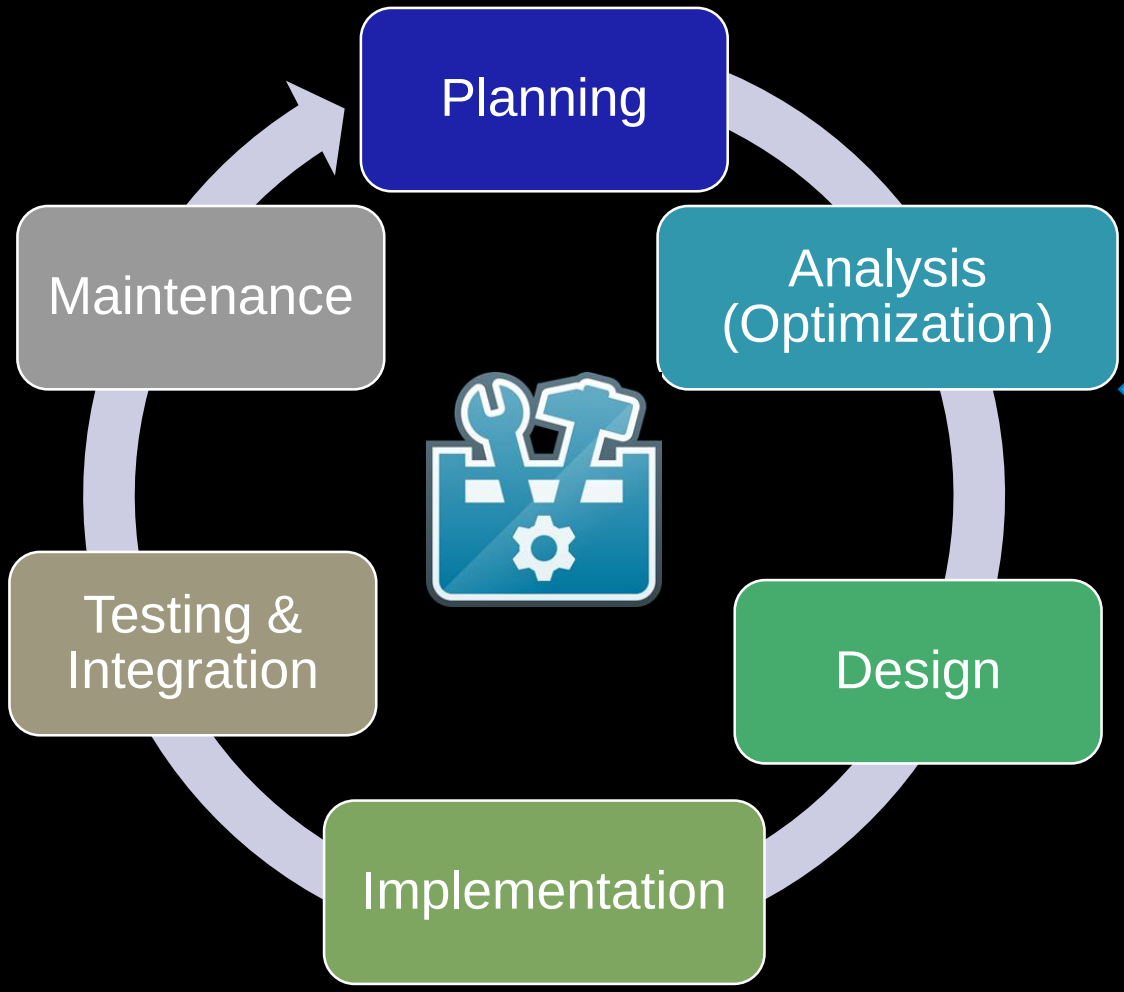
Multiplatform SDK

Improving the Developer Experience

Enterprise Software Developer

Customization Requests

Develop Own Libraries + resources + Solution (++Time ++$$)



Easy Integration to Solution (--Time --$$)

Multiplatform SDK

Improving the Developer Experience

WHY OPTIMIZATION

- Different Levels of Optimization
  - Technology Adoption & Harmonization
  - Complexity Absorption
  - Architecture & Logical Implementation
  - Customer Base
  - Support

- Stage of Engineering Process
- Is the process of modifying a system to make some features of it work more efficiently or use fewer resources.

Wikipedia

Powered by Link-OS

Technology Adoption

Card & Receipt/Label Printer - Hardware Basics

Common Aspects

- Thermal Transfer
- Print Job
- Communication
- Templates
- Status Printer
- Image printing

Powered by Link-OS

Link-OS Multiplatform SDK

Technology Adoption - Case Study

Instance & Virtualization

- Link-OS Operations
  - Printing
  - Profile (Create, Restore, Clone)
  - Configuration
  - Errors/Alerts

Utility Tasks

- Files transference (in/out)
  - Storage Image
  - Storage file
- Files conversion (in/out)
  - Graphics
  - Download Fonts
  - Convert Font
  - Convert Graphic (CPCL/ZPL)

Features Customization

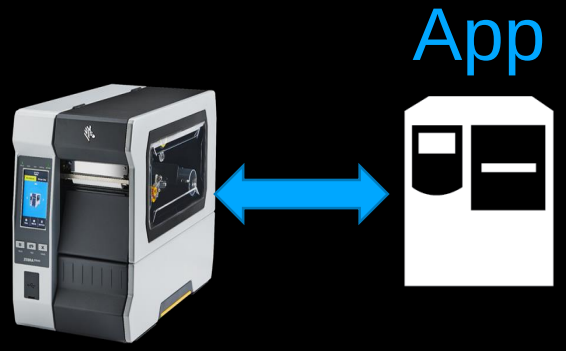
- Internal Environment to develop customized apps (ZBI)

Link-OS Multiplatform SDK

Technology Adoption - Case Study

Instance & Virtualization

- Print




```
import com.zebra.sdk.comm.Connection;
import com.zebra.sdk.comm.TcpConnection;
```

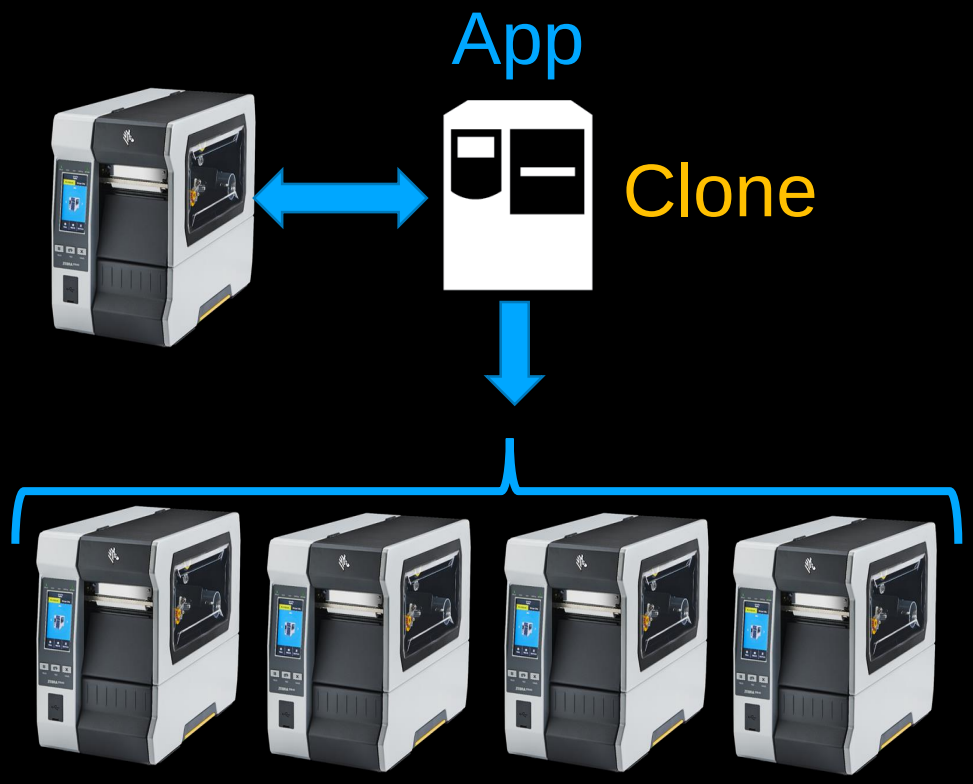
```
// Instantiate connection for ZPL TCP port at given address
 Connection thePrinterConn = new TcpConnection(theIpAddress,
                                               TcpConnection.DEFAULT_ZPL_TCP_PORT);
 try {
    // Open the connection - physical connection is established here.
    thePrinterConn.open();

    // This example prints "This is a ZPL test." near the top of the label.
    String zplData = "^XA^FO20,20^A0N,25,25^FDThis is a ZPL test.^FS^XZ";

    // Send the data to printer as a byte array.
    thePrinterConn.write(zplData.getBytes());
 } catch (ConnectionException e) {
    // Handle communications error here.
    e.printStackTrace();
 } finally {thePrinterConn.close();}
```

Link-OS Multiplatform SDK

Technology Adoption - Case Study

Instance & Virtualization

- Profile




```
import com.zebra.sdk.printer.ZebraPrinterLinkOs;
```

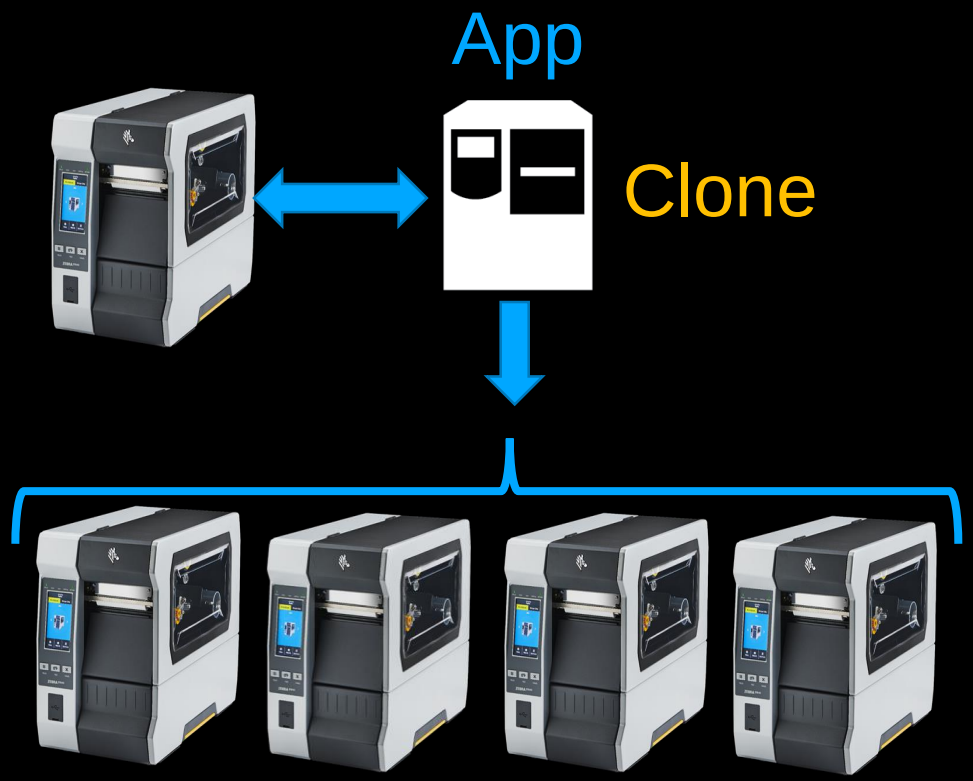
```
if (linkOsPrinter != null) {
    System.out.println("Start profile operation");
    switch (whatToDo) {
    case create_profile:
        linkOsPrinter.createProfile(pathToProfile);
        break;
    case restore_backup:
        linkOsPrinter.loadBackup(pathToProfile);
        break;
    case clone_a_printer:
        linkOsPrinter.loadProfile(pathToProfile);
        break;
    }
```

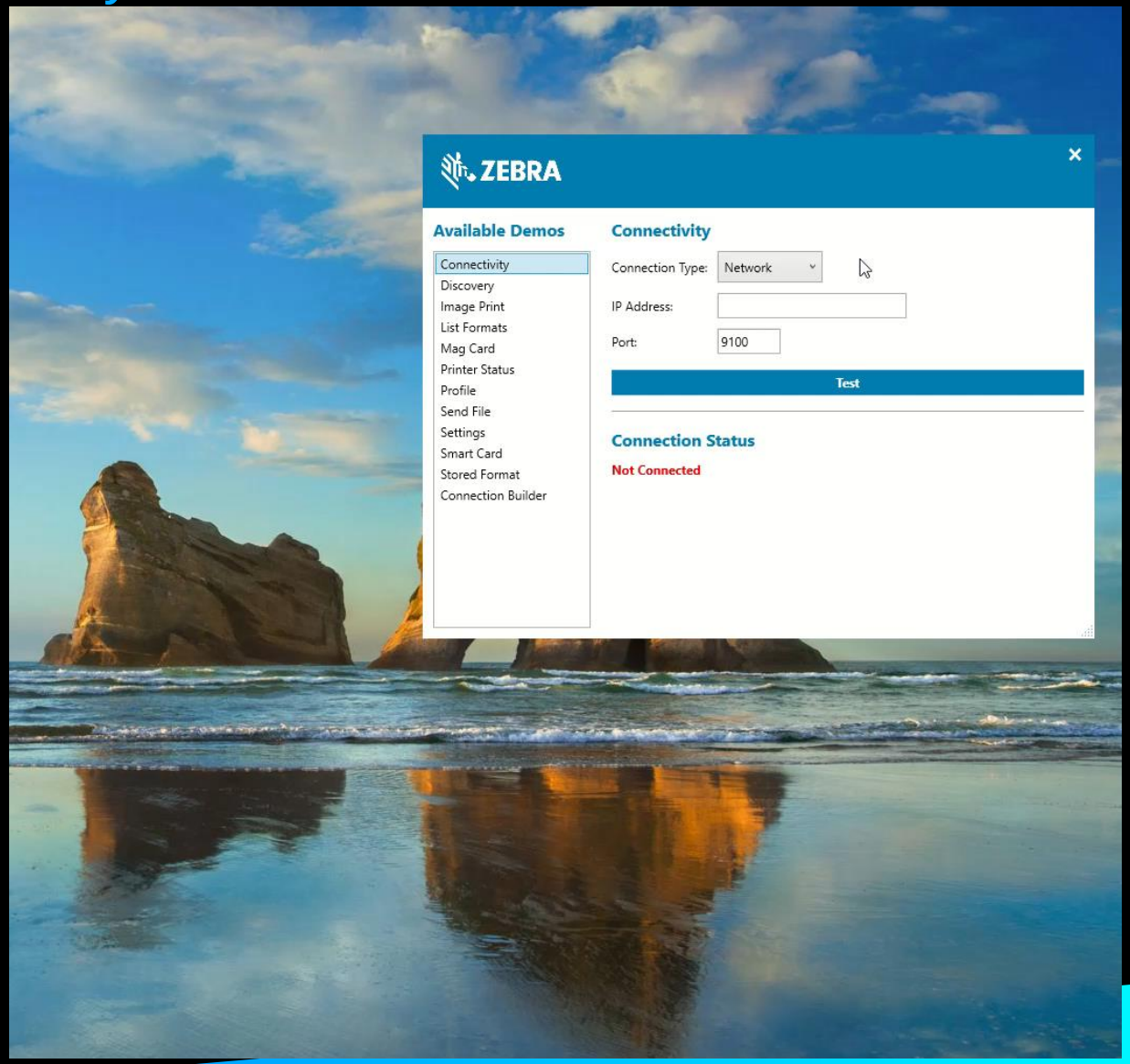
Link-OS Multiplatform SDK

Technology Adoption - Case Study

Instance & Virtualization

- Profile





Link-OS Multiplatform SDK

Technology Adoption - Case Study

Instance & Virtualization

- Configuration

```
import com.zebra.sdk.printer.SGD;
```

App

```
printerConncted = true;

if(dataSelector == false) {
    //get data here
    mediaType = SGD.GET("media.type", connection);
    ribbonEnabled = SGD.GET("ezpl.print_method", connection);
    mediaSpeed = SGD.GET("media.speed", connection);
    printedMediaLength = SGD.GET("odometer.total_print_length", connection);
    printedLabels = SGD.GET("odometer.total_label_count", connection);
    odometerMarkerCount = SGD.GET("odometer.media_marker_count", connection);
}else{
    //set data here
    SGD.SET("media.speed",mediaSpeed, connection);
}
```

Link-OS Multiplatform SDK

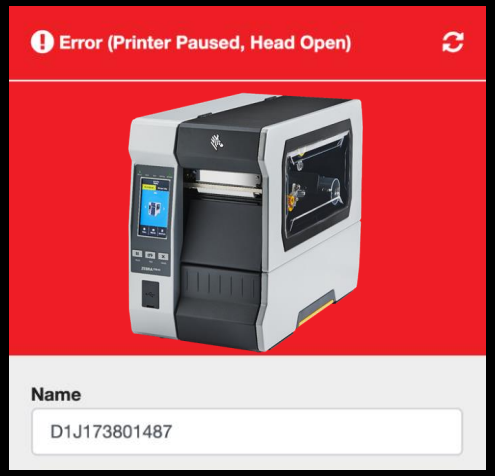
Technology Adoption - Case Study

Instance & Virtualization

- Error/Status

```
import com.zebra.sdk.printer.PrinterStatus;
```

App




```
try {
    connection.open();
    ZebraPrinter printer = ZebraPrinterFactory.getInstance(connection);
     PrinterStatus printerStatus = printer.getCurrentStatus();
    if (printerStatus.isReadyToPrint) {
        System.out.println("Ready To Print");
    } else if (printerStatus.isPaused) {
        System.out.println("Cannot Print because the printer is paused.");
    } else if (printerStatus.isHeadOpen) {
        System.out.println("Cannot Print because the printer head is open.");
    } else if (printerStatus.isPaperOut) {
        System.out.println("Cannot Print because the paper is out.");
    } else {
        System.out.println("Cannot Print.");
    }
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (ZebraPrinterLanguageUnknownException e) {
    e.printStackTrace();
} finally {connection.close();}
```

Link-OS Multiplatform SDK

Technology Harmonization - Case Study

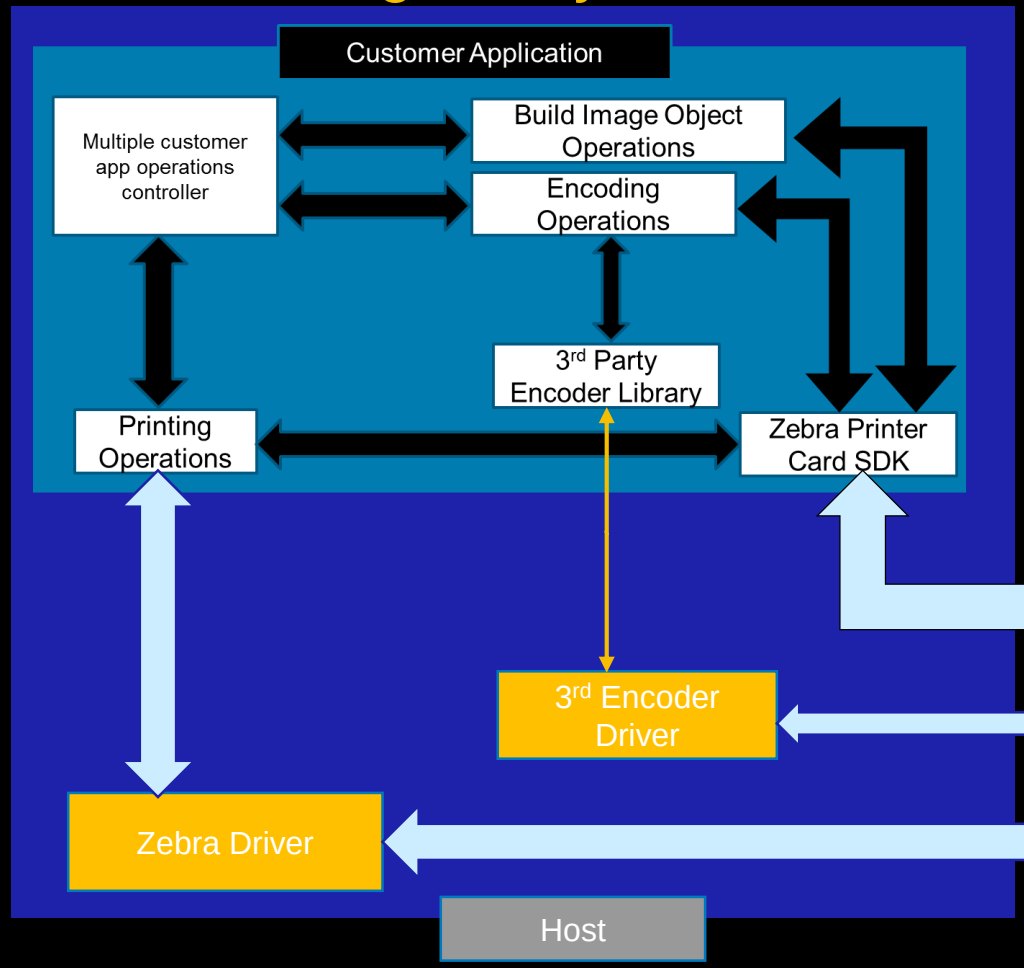
Card & Label/Receipt Printer Ecosystems

- Devices Interaction
  - Multiple Hosts & Peripherals
- APPs interaction
  - Multiple Drivers
- APIs interaction
  - Multiple SDKs
- Hardware Technology Integration
  - Multiple Sensors (IOT)
  - RFID
  - NFC
  - Multiple Cameras
  - Multiple communication interfaces (WIFI, LIFI,BT, BLE, WebSockets, etc)

Link-OS Multiplatform SDK

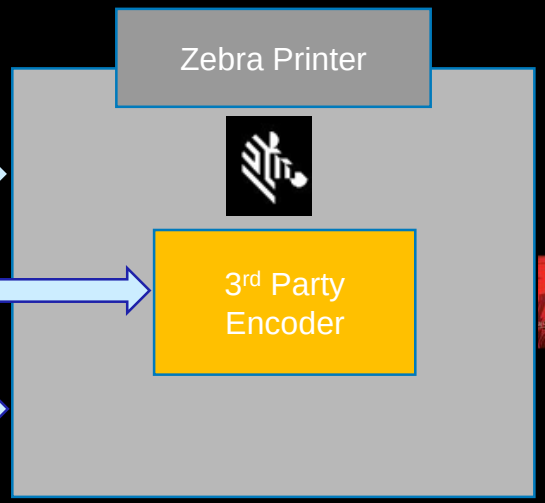
Technology Harmonization - Case Study

Card Printing Ecosystem



Zebra Printer Card SDK (LIB)

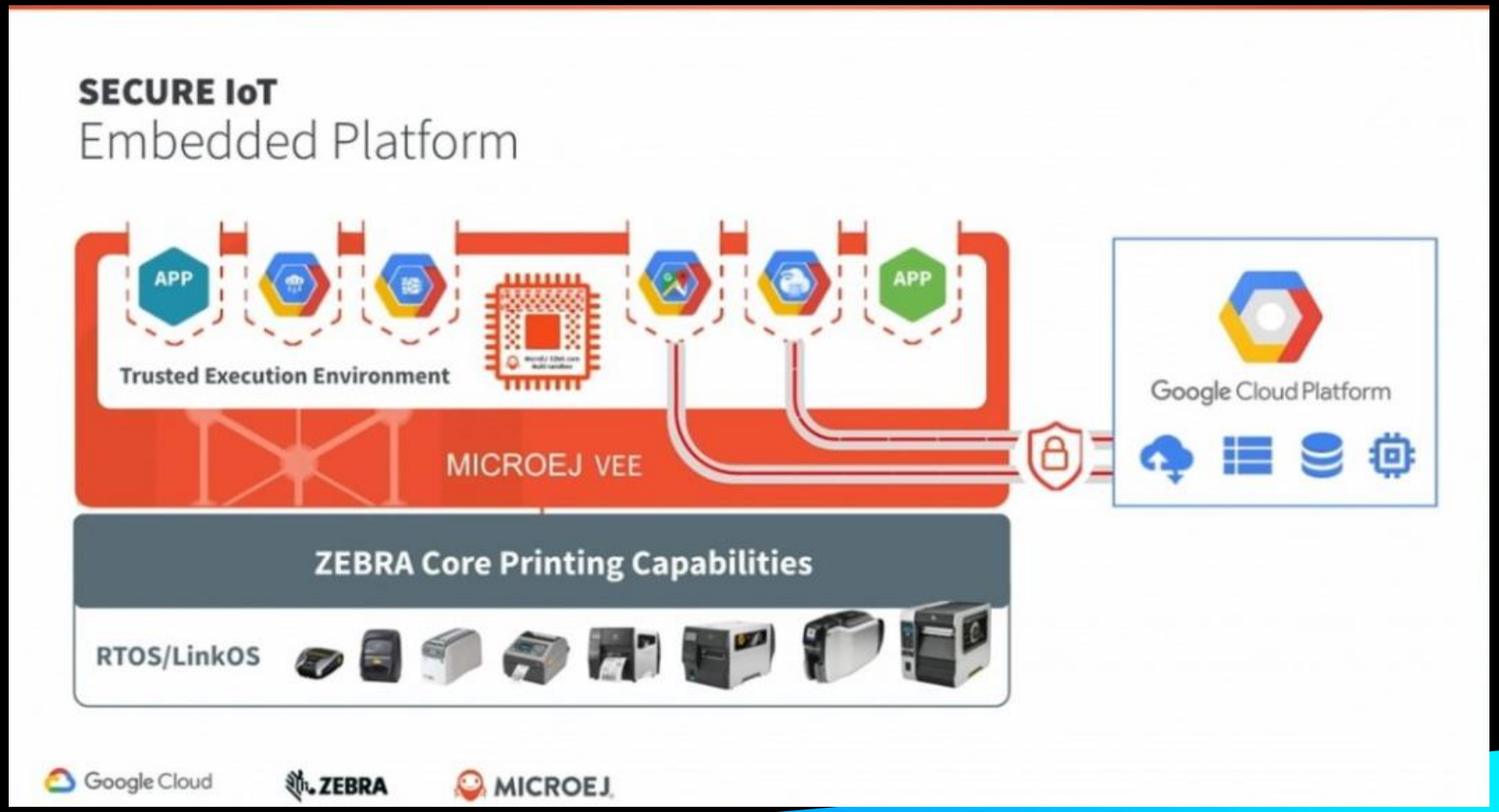
- ❑ Printing Operations
  - Bidirectional communication (Ethernet, WiFi, USB)
  - Printer settings
  - Print Job operations
- ❑ Encoding Operations
  - Magnetic encoding
  - Set the card on position to be encoded (smartcards)
- ❑ Image Operations
  - Build the object for the printer based on Ribbon's layers



Link-OS Multiplatform SDK

Technology Harmonization - Case Study

Zebra Printer Programming Ecosystem (ZBI, Java)



Link-OS Multiplatform SDK

Technology Harmonization - Case Study

NFC Tag Integration

Print Touch

NFC Tag Embedded in Link-OS printers with the icon

Contains printer info, stored as NDEF record - structured as a URL:

http://www.zebra.com/apps/r/nfc?mB=aabbccddeeff&mW=000000000000&mE=aabbccddeeff&c=123456789012345&s=12345678901234&v=0

Field	Content	Delimiter
"URL" Header:	http://www.zebra.com/apps/r/nfc	?
Bluetooth MAC Address:	mB=aabbccddeeff	&
WLAN MAC Address:	mW=000000000000	&
Ethernet MAC Address:	mE=aabbccddeeff	&
Printer Configurator (15 digits):	c=123456789012345	&
Printer Serial Number (14 digits):	s=12345678901234	&
Zebra "URL Record" Version:	v=0	(none)

See source code examples in Print Station and developer demos

Two ways to use:

- Trigger app launch
- Capture NFC read only when app open

Link-OS Multiplatform SDK

Technology Harmonization - Case Study

NFC Tag Integration

Print Touch

```
private void processNfcScan(Intent intent, final Context context) {
    Parcelable[] scannedTags = intent.getParcelableArrayExtra(NfcAdapter.EXTRA_NDEF_MESSAGES);
    if (scannedTags != null && scannedTags.length > 0) {
        NdefMessage msg = (NdefMessage) scannedTags[0];
        final String payload = new String(msg.getRecords()[0].getPayload());
        new AsyncTask<Void, Void, Void>() {
            protected Void doInBackground(Void... params) {
                try {
                    UrlPrinterDiscoverer.findPrinters(payload, new DiscoveryHandler() {
                        public void foundPrinter(DiscoveredPrinter printer) {
                            try {
                                Connection conn = printer.getConnection();
                                conn.open();
                                ZebraPrinterFactory.getInstance(conn).printConfigurationLabel();
                                conn.close();
                            } catch (Exception e) {
                                e.printStackTrace();
                            }
                        }
```

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

Interfaces simplification

- Communication Interface
- UX Interface - Templates
  - (Card & Label/Receipt)

Multitask processing

- Emulating firmware functions through classes/methods

Faster Device Discovery

- Filtering techniques
- Device Identification

Open Bi-directional User Channel

- Optimization of connection timeouts

Security

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

APIs Simplification for communication interface - Standardization

Link-OS SDK main classes for BT/BLE

- Discovery
  - BluetoothDiscoverer (same TCP)
  - DiscoveredPrinter
  - DiscoveredPrinterBluetooth (same TCP)
- Communication
  - Connection
    - BluetoothConnection
    - BluetoothLeConnection
    - TcpConnection
    - UsbConnection
  - ZebraPrinter

ZEBRA

Android SDK main classes for BLE

- Discovery
  - DeviceScanActivity
  - Handler
- Communication
  - BluetoothAdapter
  - BluetoothDevice
  - BluetoothGatt
  - BluetoothGattCallBack
  - BluetoothGattCharasteristic
  - BluetoothGattDescriptor
  - BluetoothGattService
  - BluetoothManager
  - BluetoothProfile
  - BroadcastReceiver

Android SDK main classes for BT

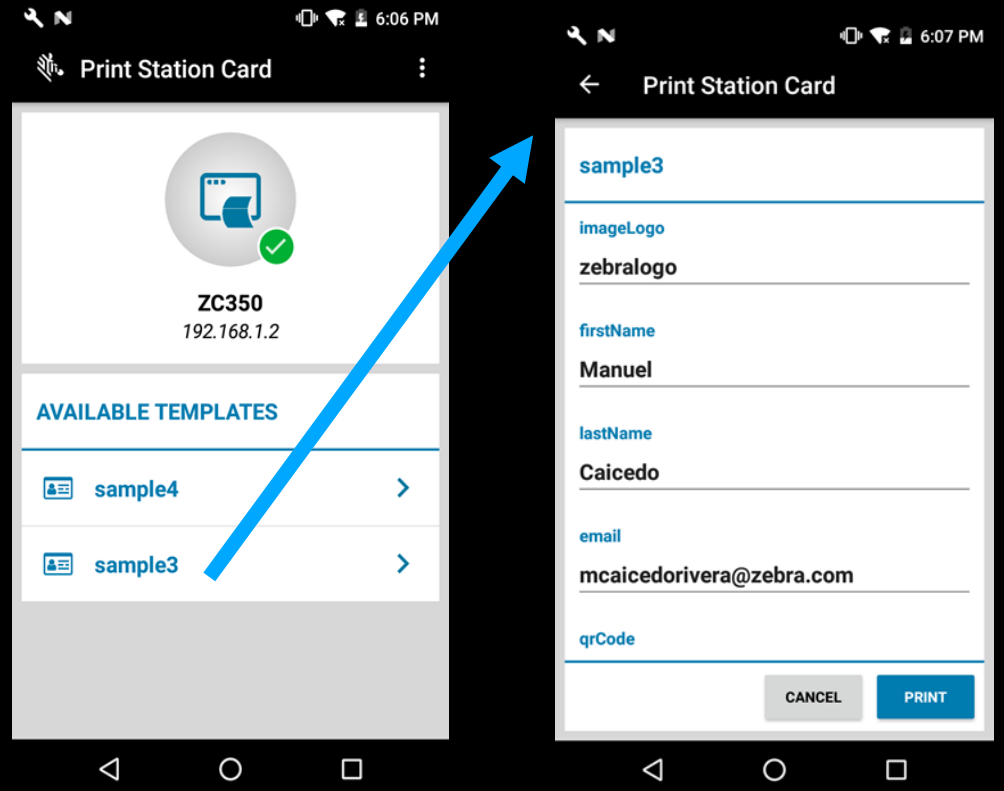
- Discovery
  - BroadCastReciever
  - IntentFilter
  - registerReceiver
  - unregisterReceiver
- Communication
  - BluetoothAdapter
  - BluetoothDevice
  - BluetoothSocket

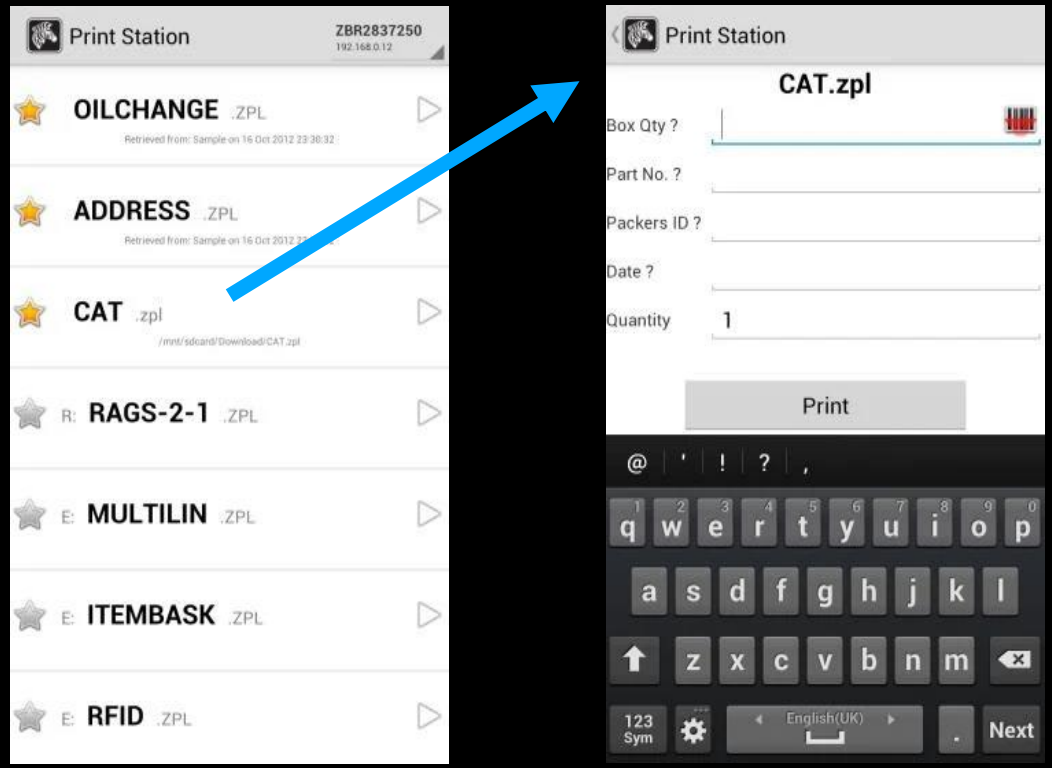
Developers

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

UX Interface Simplification - Templates





Link-OS Multiplatform SDK

Complexity Absorption - Case Study

UX Interface Simplification - Templates

XML Data Document:

```
<data>
  <image1>NameOfStoredImage_1</image1>
  <image2>NameOfStoredImage_2</image2>
  <firstName>Richard</firstName>
  <lastName>Smith</lastName>
  <barcodeData>123456</barcodeData>
  <track1Data>TEMPLATE TRACK 1 DATA</track1Data>
  <track2Data>9879654321</track2Data>
  <track3Data>11022033044055066</track3Data>
</data>
```

Instructions

Layer

Barcode

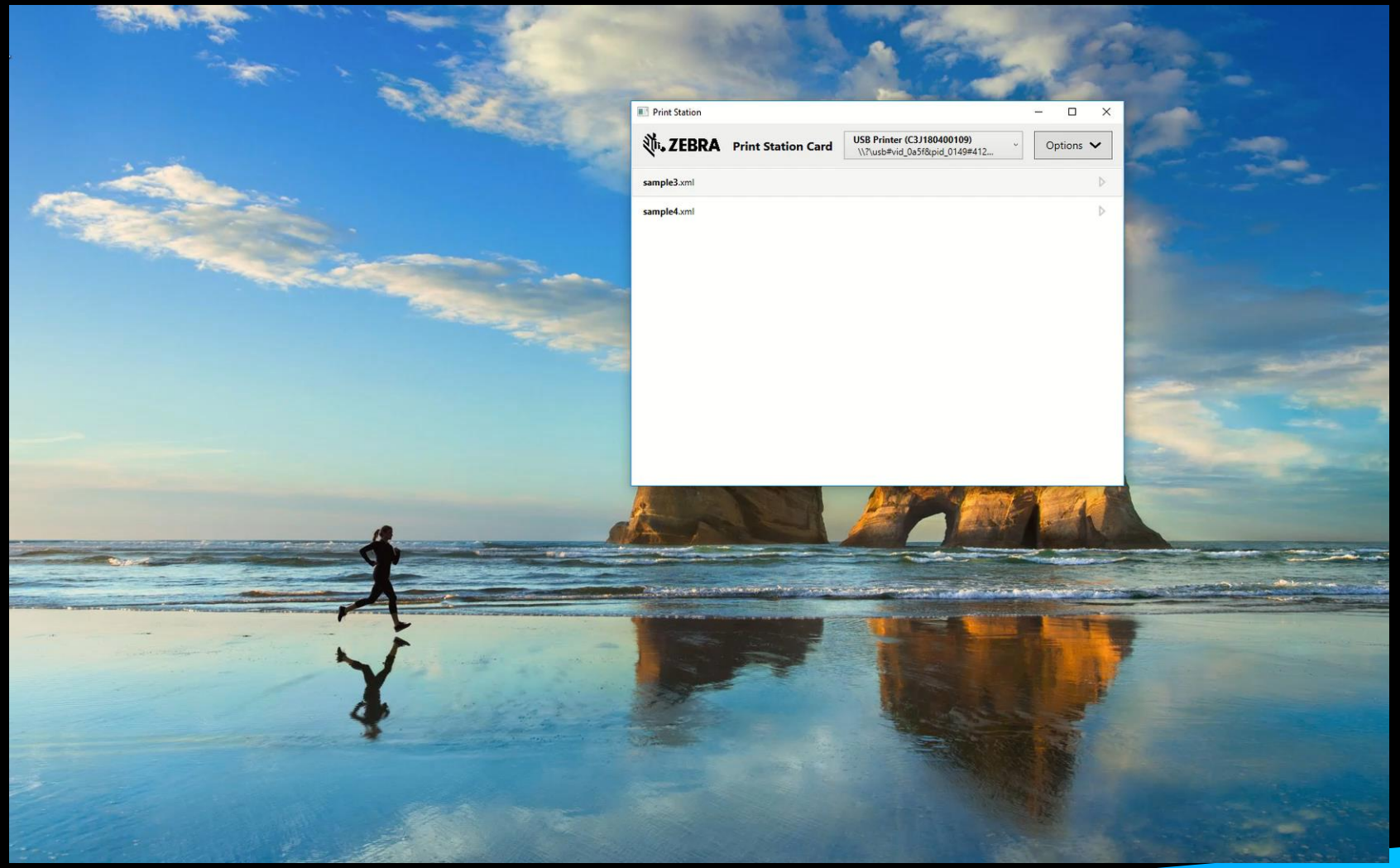
MagData

```
<?xml version="1.0" encoding="utf-8"?>
<template name="Template" card_type="2" card_thickness="30" source="feeder" destination="eject" delete="">
 <fonts>
  <font id="1" name="arial" size="12" bold="no" italic="no" underline="no"/>
  <font id="2" name="calibri" size="14" bold="no" italic="yes" underline="no"/>
 </fonts>
 <sides>
  <side name="front" rotation="0" sharpness="">
   <print_types>
    <print_type type="color" fill="">
     <graphic field="image1" format="bmp" opacity="100" width="0" height="0" x="100" y="100" delete="yes"/>
     <graphic field="image2" format="bmp" width="610" height="325" x="400" y="50"/>
    </print_type>
    <print_type type="mono" fill="">
     <text field="firstName" font_id="1" width="0" height="0" x="100" y="250" angle="0" color="0xFFFFFF" alignment="left"/>
     <text field="lastName" font_id="2" width="0" height="0" x="100" y="310" angle="0" color="0xFFFFFF" alignment="left"/>
    </print_type>
   </print_types>
  </side>
  <side name="back" orientation="landscape" rotation="0" sharpness="normal">
   <print_types>
    <print_type type="mono" fill="">
     <text field="firstName" font_id="1" width="0" height="0" x="50" y="375" angle="0" color="0xFFFFFF" alignment="left"/>
     <barcode order_id="1" field="barcodeData" height="110" width="200" x="820" y="250" code="code39" rotation="90"
      show_text="yes" quiet_zone_width="0"/>
     <line x1="55" y1="440" x2="220" y2="440" thickness="4" color="0xFFFFFF"/>
    </print_type>
   </print_types>
   <magdata format="iso" coercivity="high" verify="yes">
    <track field="track1Data" number="1" format="ascii"/>
    <track field="track2Data" number="2" format="ascii"/>
    <track field="track3Data" number="3" format="ascii"/>
   </magdata>
  </side>
 </sides>
</template>
```

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

UX Interface Simplification - Templates



Link-OS Multiplatform SDK

Complexity Absorption - Case Study

UX Interface Simplification - Templates

```
@Override
protected Void doInBackground(Void... params) {
    Connection connection = null;
    ZebraCardPrinter zebraCardPrinter = null;

    try {
        connection = printer.getConnection();
        connection.open();

        zebraCardPrinter = ZebraCardPrinterFactory.getInstance(connection);

        TemplateJob templateJob = zebraCardTemplate.generateTemplateJob(templateName, variablesData);
        jobId = zebraCardPrinter.printTemplate(quantity, templateJob);
    } catch (Exception e) {
        exception = e;
    } finally {
        ConnectionHelper.cleanUpQuietly(zebraCardPrinter, connection);
    }

    return null;
}
```

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

UX Interface Simplification – Templates

```
^XA^DFXmlExamp.zpl^FS
^A0N,100,100^FO100,100^FN1"Name"^FS
^A0N,100,100^FO100,200^FN2"Street"^FS
^A0N,100,100^FO100,300^FN3"City"^FS
^A0N,100,100^FO100,400^FN4"State"^FS
^A0N,100,100^FO100,500^FN5"Zip"^FS
^XZ
```

```
String sampleXmlData =
        "<?xml version=\"1.0\" encoding=\"UTF-8\"?>"
                + "<file _FORMAT=\"XmlExamp.zpl\">"
                + " <label>\n"
                + "     <variable name=\"Name\">John Smith</variable>"
                + "     <variable name=\"Street\">1234 Anystreet</variable>"
                + "     <variable name=\"City\">Anycity</variable>"
                + "     <variable name=\"State\">Anystate</variable>"
                + "     <variable name=\"Zip\">12345</variable>"
                + "  </label>\n"
                + " <label>\n"
                + "     <variable name=\"Name\">Jane Doe</variable>"
                + "     <variable name=\"Street\">5678 Anyroad</variable>"
                + "     <variable name=\"City\">Anytown</variable>"
                + "     <variable name=\"State\">Anystate</variable>"
                + "     <variable name=\"Zip\">67890</variable>"
                + "  </label>\n"
                + "</file>";
```

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

```
String destinationDevice = "192.168.1.32";
String templateFilename = "c:\\XmlPrinterExampleTemplate.zpl";
String defaultQuantityString = "1";
boolean verbose = true;
```

UX Interface Simplification – Templates

```
^XA^DFXmlExamp.zpl^FS
^A0N,100,100^FO100,100^FN1"Name"^FS
^A0N,100,100^FO100,200^FN2"Street"^FS
^A0N,100,100^FO100,300^FN3"City"^FS
^A0N,100,100^FO100,400^FN4"State"^FS
^A0N,100,100^FO100,500^FN5"Zip"^FS
^XZ
```

```
try {
    ByteArrayOutputStream outputDataStream = new ByteArrayOutputStream();
    XmlPrinter.print(destinationDevice,
                     getSampleXmlData(),
                     templateFilename,
                     defaultQuantityString,
                     outputDataStream,
                     verbose);
    System.out.println(outputDataStream.toString());
} catch (Exception e) {
    e.printStackTrace();
}
```

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

Multitask Processing – Firmware Emulation

ZXP1/ZXP3 templates to emulate the same process through ZXP 7/9.

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

Faster Device Discovery

Filtering techniques

Only to recognize Zebra Devices during discovery

Device Identification

ID profile to show type/model device during discovery

- TCP/IP (6 fields)
- BT (MAC, Friendly Name)

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

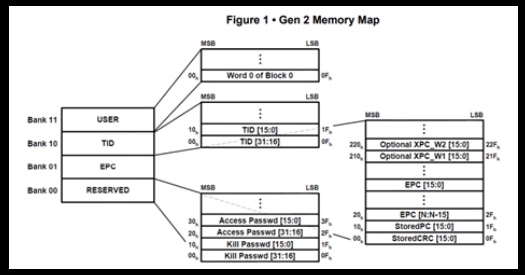
Open bi-directional raw channel

Example • The following code:

```
^XA
.
.
.
^FH_^HV0,8,EPC[,]_0D_0A,L^FS
^PQ2
^XZ
```

Would return data such as this:

```
EPC[12345678]
EPC[55554444]
```




```
// Open the connection - physical connection is established here.
thePrinterConn.open();
 // This example prints "This is a ZPL test." near the top of the label.
String zplData = "^XA^FO20,20^A0N,25,25^FRFID ZPL test.^FS^FH_^HV0,8,EPC[,]_0D_0A,L^FS^PQ2^XZ";

byte [] EPC_data = null;

// Send and receive the data to printer as a byte array.
EPC_data = thePrinterConn.sendAndWaitForResponse(zplData.getBytes(),340, 500, null);

// Close the insecure connection to release resources.
thePrinterConn.close();
```

Optimization of connection timeouts

For example, when bi-directional communication requires timeouts to wait for values to be returned (RFID)

Link-OS Multiplatform SDK

Complexity Absorption - Case Study

PRINTSECURE

Printer Security Made Simple
Embedded in Link-OS®

Security Implementations

Implementation simplified of certificates for configuration through WIFI

```
try {
   String caContent = zebraP12Info.getCaContent();
   String clientCertificateContent = zebraP12Info.getCertificateContent();
   String encryptedPrivateKeyContent = zebraP12Info.getEncryptedPrivateKeyContent
                                       (privateKeyEncryptionPassword, p12FilePassword);

   conn = new DriverPrinterConnection("myPrinterDriverName");
   conn.open();
   ZebraPrinterLinkOs linkosPrinter = ZebraPrinterFactory.getLinkOsPrinter(conn);

   linkosPrinter.storeFileOnPrinter(clientCertificateContent.getBytes(),
                                    ZebraCertificateInfo.CLIENT_CERT_NRD_PRINTER_FILE_NAME);
   linkosPrinter.storeFileOnPrinter(caContent.getBytes(),
                                    ZebraCertificateInfo.CA_CERT_NRD_PRINTER_FILE_NAME);
   linkosPrinter.storeFileOnPrinter(encryptedPrivateKeyContent.getBytes(),
                                    ZebraCertificateInfo.CLIENT_PRIVATE_KEY_NRD_PRINTER_FILE_NAME);
   linkosPrinter.setSetting("wlan.private_key_password", privateKeyEncryptionPassword);
} catch (Exception e) {
   e.printStackTrace();
} finally {
   if (conn != null) {
      conn.close();
   }
}
```

Link-OS Multiplatform SDK

Customer Base - Case Study

Developer/User Preferences

- IDE Integration
  - Library Integration through external repositories
- Programming Languages
  - C#, Java, Android, IOS, (JavaScript)
- Platforms
  - Windows, Linux/Unix, Cloud

Use Cases Implementation

- Solve Specific Customer Needs
  - Multichannel communication
  - Multilayer printing

On-Demand Market Requests

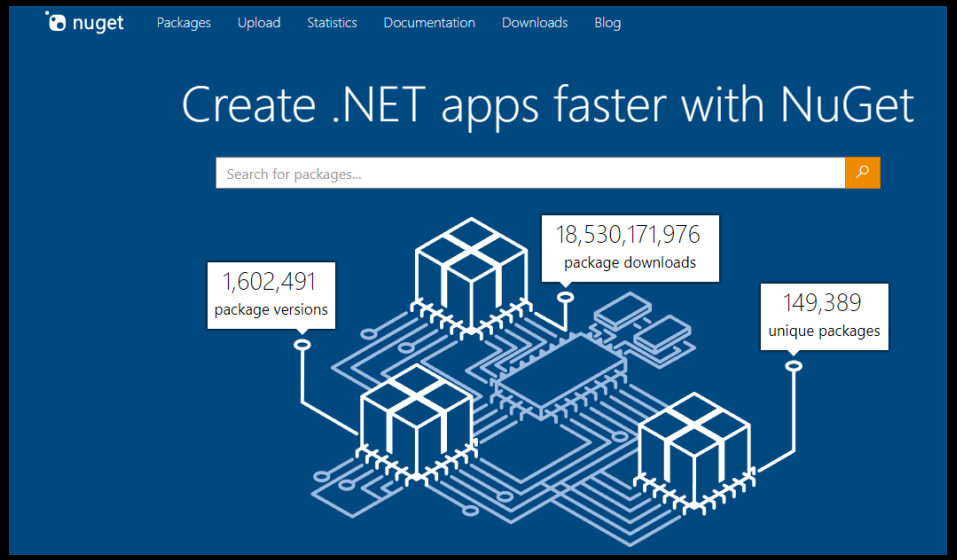
- Market Disruption
  - New Technologies
- Market Leaders
  - Policies

Link-OS Multiplatform SDK

Customer Base - Case Study

Library Integration through external repositories

- NuGet Repositories for Visual Studio



Zebra.Printer.SDK 2.15.2634

The Zebra Link-OS SDK provides a powerful set of APIs enabling creation of apps that take full advantage of the printer's operating system features including, connectivity, printing and management tasks.

Package Manager | .NET CLI | PackageReference | Paket CLI

```
PM> Install-Package Zebra.Printer.SDK -Version 2.15.2634
```

Zebra.Printer.Card.SDK 2.15.2634

The Zebra Link-OS SDK provides a powerful set of APIs enabling creation of apps that take full advantage of the printer's operating system features including connectivity, printing and management tasks.

Package Manager | .NET CLI | PackageReference | Paket CLI

```
PM> Install-Package Zebra.Printer.Card.SDK -Version 2.15.2634
```

Link-OS Multiplatform SDK

Customer Base - Case Study

Use Cases Implementation

Multichannel Implementation to reduce time for batch printing

```
Connection thePrinterConn = new MultichannelTcpConnection(theIpAddress,
                                MultichannelTcpConnection.DEFAULT_MULTICHANNEL_PRINTING_PORT,
                                MultichannelTcpConnection.DEFAULT_MULTICHANNEL_STATUS_PORT);
```

App

```
// Opens the connection - physical connection is established here.
thePrinterConn.open();

// Creates a Link-OS printing with the given connection
ZebraPrinterLinkOs linkOsPrinter = ZebraPrinterFactory.getLinkOsPrinter(thePrinterConn);

// This is sent over the printing channel (9100 by default)
// until label format is completely sent.
String labelFormatStartCommand = "^XA";
linkOsPrinter.sendCommand(labelFormatStartCommand);

String labelBody = "^FO20,20^A0N,25,25^FDThis is a ZPL test.^FS";
linkOsPrinter.sendCommand(labelBody);

// This is sent over the status channel (9200 by default)
// and will return immediately even though the printing channel is in use.
PrinterStatus status = linkOsPrinter.getCurrentStatus();

System.out.println("The printer PAUSED state is : " + status.isPaused);

// Send the end of label command to finish and print the label.
String labelFormatEndCommand = "^XZ";
linkOsPrinter.sendCommand(labelFormatEndCommand);
thePrinterConn.close();
```

Link-OS Multiplatform SDK

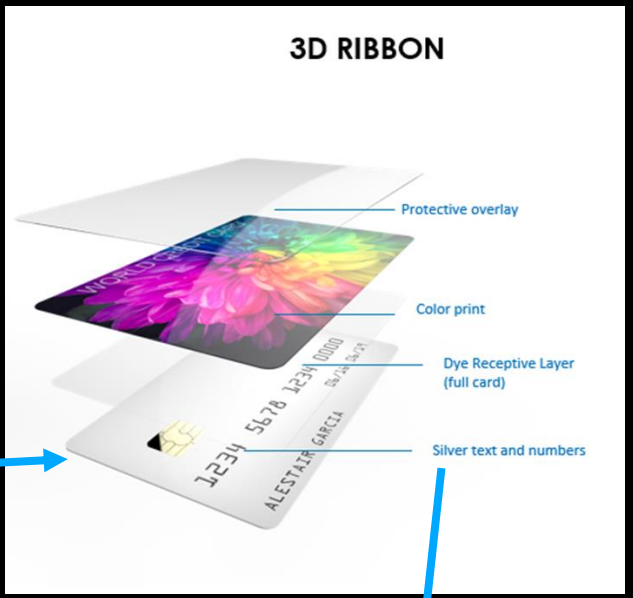
Customer Base - Case Study

Use Cases Implementation

Card Multilayer printing to optimize use cases from customer implementation

```
if (frontSideOverlayImageData != null) {
    graphics.initialize( i: 0,  i1: 0, OrientationType.Landscape, PrintType., Color.WHITE);
    graphics.drawImage(frontSideOverlayImageData,  i: 0,  i1: 0,  i2: 0,
    graphicsData.add(buildGraphicsInfo(graphics.createImage(), CardS
    graphics.clear();
} else if (frontSideImageUriMap.containsKey(PrintType.Overlay)) {
    graphicsData.add(buildGraphicsInfo( zebraCardImage: null, CardSide.
    graphics.clear();
}

if (backSideImageData != null) {
    graphics.initialize( i: 0,  i1: 0, OrientationType.Landscape, Print
    graphics.drawImage(backSideImageData,  i: 0,  i1: 0,  i2: 0,  i3: 0,
    graphicsData.add(buildGraphicsInfo(graphics.createImage(), CardS
    graphics.clear();
}
```

```
valueOf(String name)          PrintType
Color                         PrintType
GrayDye                       PrintType
GrayGhost                     PrintType
GrayUV                        PrintType
Helper                        PrintType
Inhibit                       PrintType
MonoGhost                     PrintType
MonoK                         PrintType
MonoKHigh                     PrintType
MonoUV                        PrintType
MonoWhite                     PrintType
MonoWhiteHigh                 PrintType
Overlay                       PrintType
Pearl                         PrintType
SilverResin                   PrintType
WhiteResin                    PrintType
values()                      PrintType[]
valueOf(Class<T> enumType, String name)  T
class
```



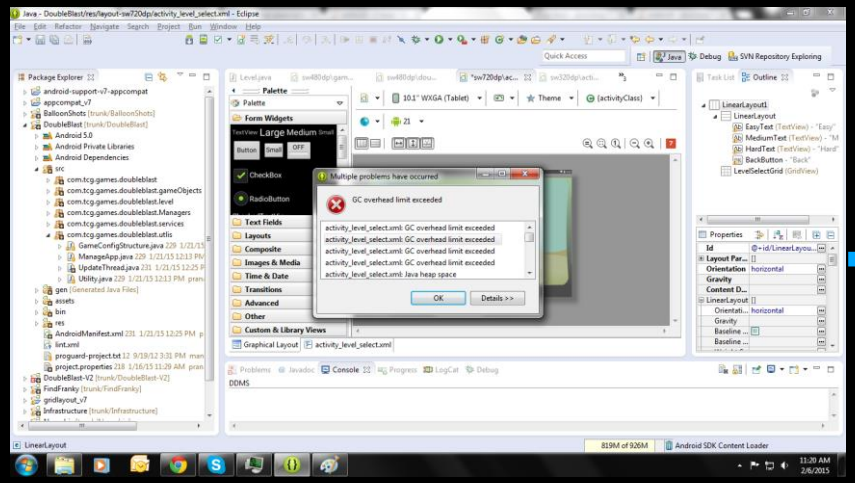

```
List<GraphicsInfo> graphicsData = new ArrayList<>();
graphicsData.add(buildGraphicsInfo(graphics.createImage(), CardSide.Front, PrintType.Overlay));
...
graphicsData.add(buildGraphicsInfo(graphics.createImage(), CardSide.Front, PrintType.Color));
...
graphicsData.add(buildGraphicsInfo(graphics.createImage(), CardSide.Front, PrintType.SilverResin));
...

jobId = zebraCardPrinter.print(printOptions.getQuantity(), graphicsData);
```

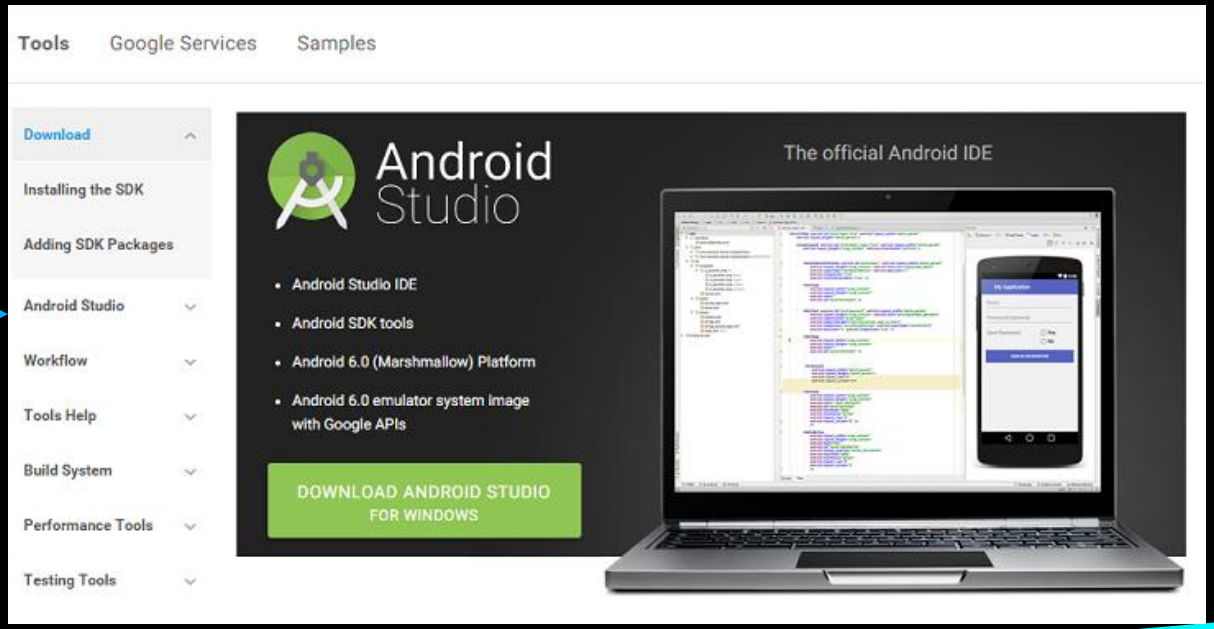
Link-OS Multiplatform SDK

Customer Base - Case Study

- On-Demand Market Requests
  - Moving from Eclipse to Android Studio
    - Gradle build automation
    - Maven to JCenter
    - Kotlin

New SDK Sample Code developed in Android Studio





Link-OS Multiplatform SDK

Architecture & Logical Implementation - Case Study

APIs Optimization

- Algorithms Optimized
- 3rd party libraries current and updated

Architecture

- Solution Architecture simplified for integration

Data

- Access/Reporting
  - Individual – SGD
  - Data Sets – JSON

Link-OS Multiplatform SDK

Architecture & Logical Implementation - Case Study

Printer Discovery
USB, Bluetooth, Network or BLE

Printer Connectivity
USB, Bluetooth, Network, Web-Sockets

Printer Status Checking
Errors, Warning, Alerts

Font Conversion
True Type to ZPL Font

Graphics Conversion
PNG & BMP to ZPL Graphics

Template Filling
Fill ZPL templates with variable data

Printer Management
Create/Send Profile/Print OS's

Command Mode
Scriptable command line

Simplified Pairing
Using The Print Touch feature (NFC tag)

Developer Demos, Sample Code & Documentation
Both Source Code and compiled demo code for commonly used features
Sample code for all major functions, Extensive API documentation that embeds within IDE platforms

Link-OS Multiplatform SDK

Architecture & Logical Implementation - Case Study

Data Access

Individual – SGD
Data Sets – JSON

```
byte[] JSon_sent = JSon_Battery.getBytes();

ZebraPrinter printer = ZebraPrinterFactory.getInstance(connection);

if (printer.getCurrentStatus().isReadyToPrint) {
    printerConncted = true;
    try {
        if (!dataSelector) {
            Json_received = connection.sendAndWaitForResponse(JSon_sent, 100, 120, null);
            Json_converted = new String(Json_received, "UTF-8");
            System.out.format(" Json captured: %s%n", Json_converted);
        } else {
            //set data here
            SGD.SET("power.sleep.enable", powerSleepEnabled, connection);
        }
    } catch (Exception e) {
        helper.showErrorDialogOnGuiThread(e.getMessage());
    }
```

```
if (Json_received != null) {
    batteryData = dataJson.jsonToMap(Json_converted);
    batteryVoltage = batteryData.get("power.voltage");
    batteryPercentFull = batteryData.get("power.percent_full");
    batteryStatus = batteryData.get("power.status");
    batteryHealth = batteryData.get("power.health");
    batteryRechargeCls = batteryData.get("power.cycle_count");
    powerSleepEnabled = batteryData.get("power.sleep.enable");
}
```

```
String JSon_Battery = "{}{\"power.voltage\":null," +
        "\"power.percent_full\":null," +
        "\"power.status\":null," +
        "\"power.health\":null," +
        "\"power.cycle_count\":null," +
        "\"power.sleep.enable\":null}";
```

Link-OS Multiplatform SDK

Support - Case Study

SDK Development & Maintenance

- SDK NPD Teams

Customer Support

- Technical Support
- Special Services – Developer Customization Groups
- Developer Relations (Advocates, advisers)

Manufacturer Recommendation Guides

- Zebra Best Practices, White Papers, Blogs

Developer Forums

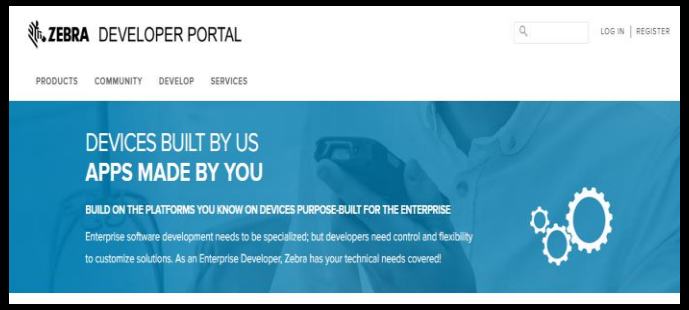
- Community support

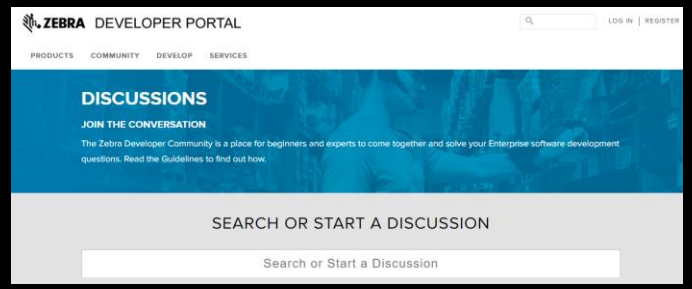
Documentation

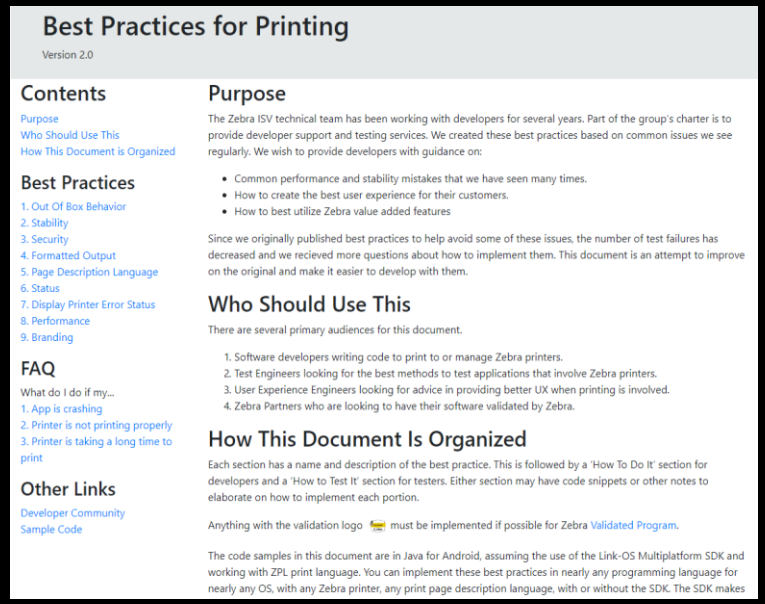
- Guides & Manuals

Link-OS Multiplatform SDK

Support - Case Study







Where to get it

SDK

- Link-OS™ Multiplatform SDK
  - **Unix & Mac OS**
  - **Windows**
  - **Linux**

Other ways to find it:

- www.zebra.com/sdk

Online Documentation

- http://techdocs.zebra.com/

Zebra Developers

Build Your Edge

Questions?

http://developer.zebra.com

Zebra Developer Community – LinkedIn Group

@ZebraDevs

ZEBRA

ZEBRA and the stylized Zebra head are trademarks of Zebra Technologies Corp., registered in many jurisdictions worldwide. All other trademarks are the property of their respective owners. ©2019 Zebra Technologies Corp. and/or its affiliates. All rights reserved.