

What is DevOps and What Can It Do?

Presented by

SAMSUNG

REDHAT SUMMIT 2016

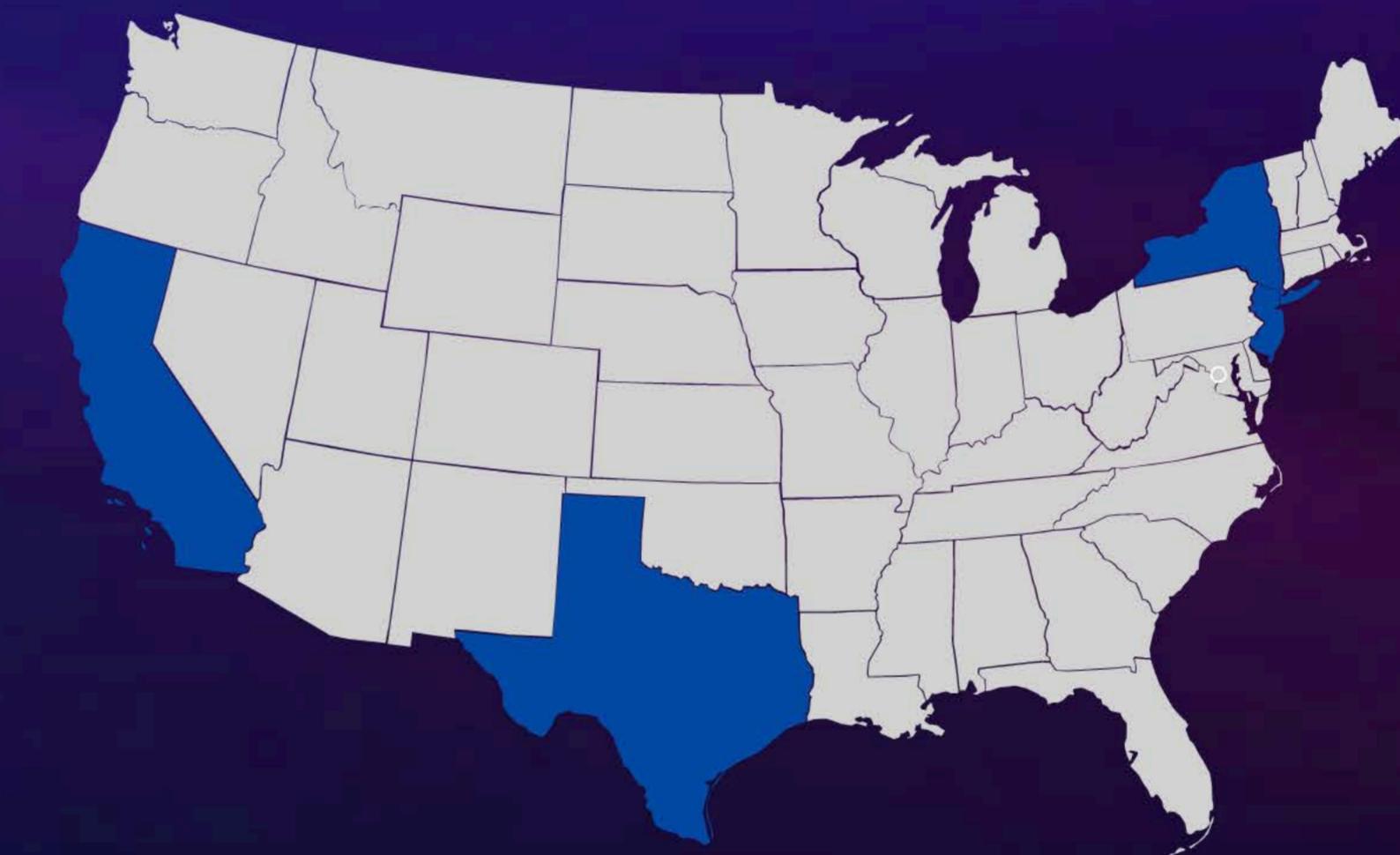


Welcome



Richard Balsewicz

Innovation Leader
Samsung Business Services



Founded Services Innovation Lab
Leads Off-the-Radar Enterprise Innovation
Most rewarding x challenging position to date
We build things people use everyday

Think Fast #1

Hacker?

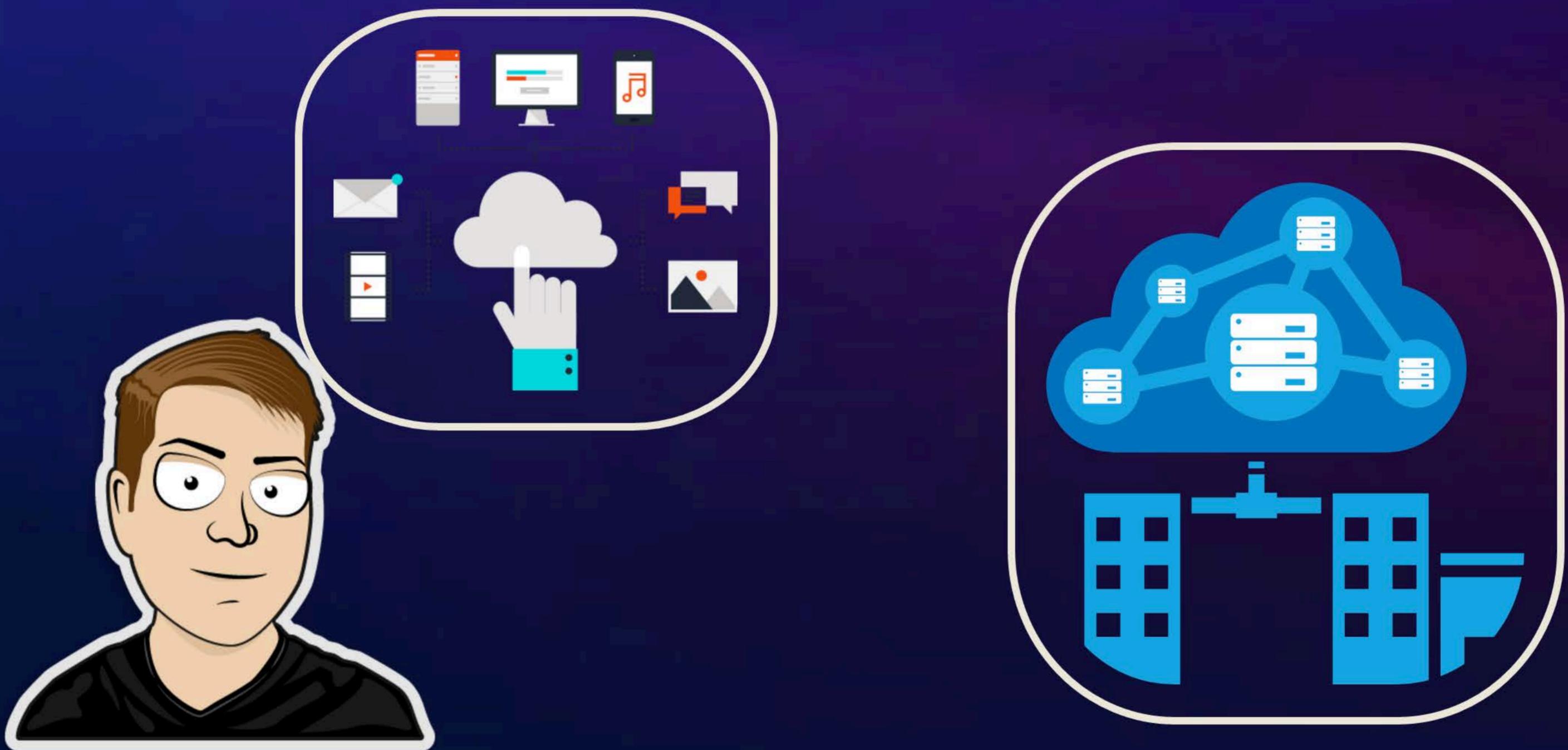
Word Association



Word Association



My Association



De-Personified View

N machines vs. Your Machine(s)

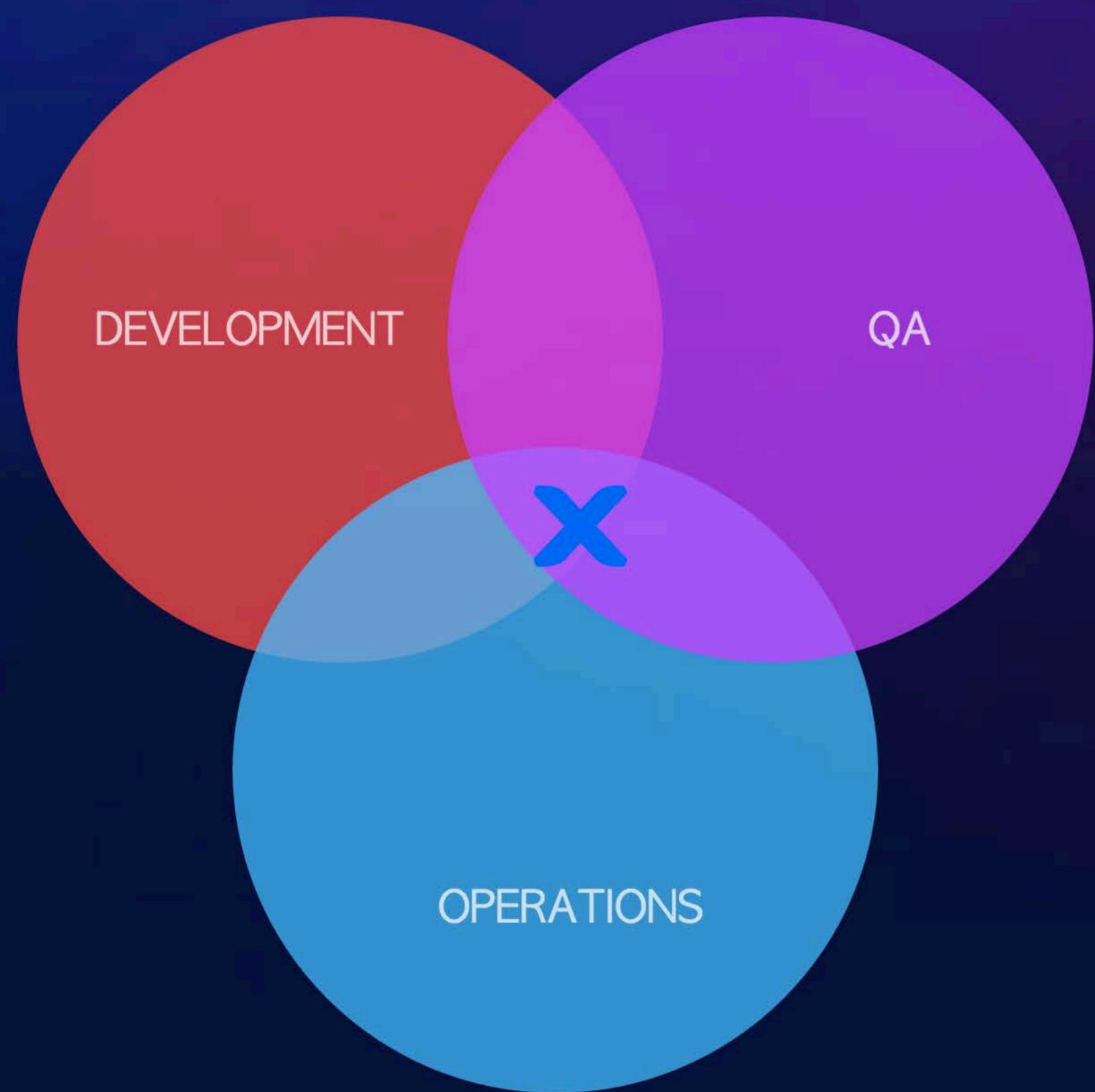
Fast will lead to Fail

Think Fast #2

DevOps?

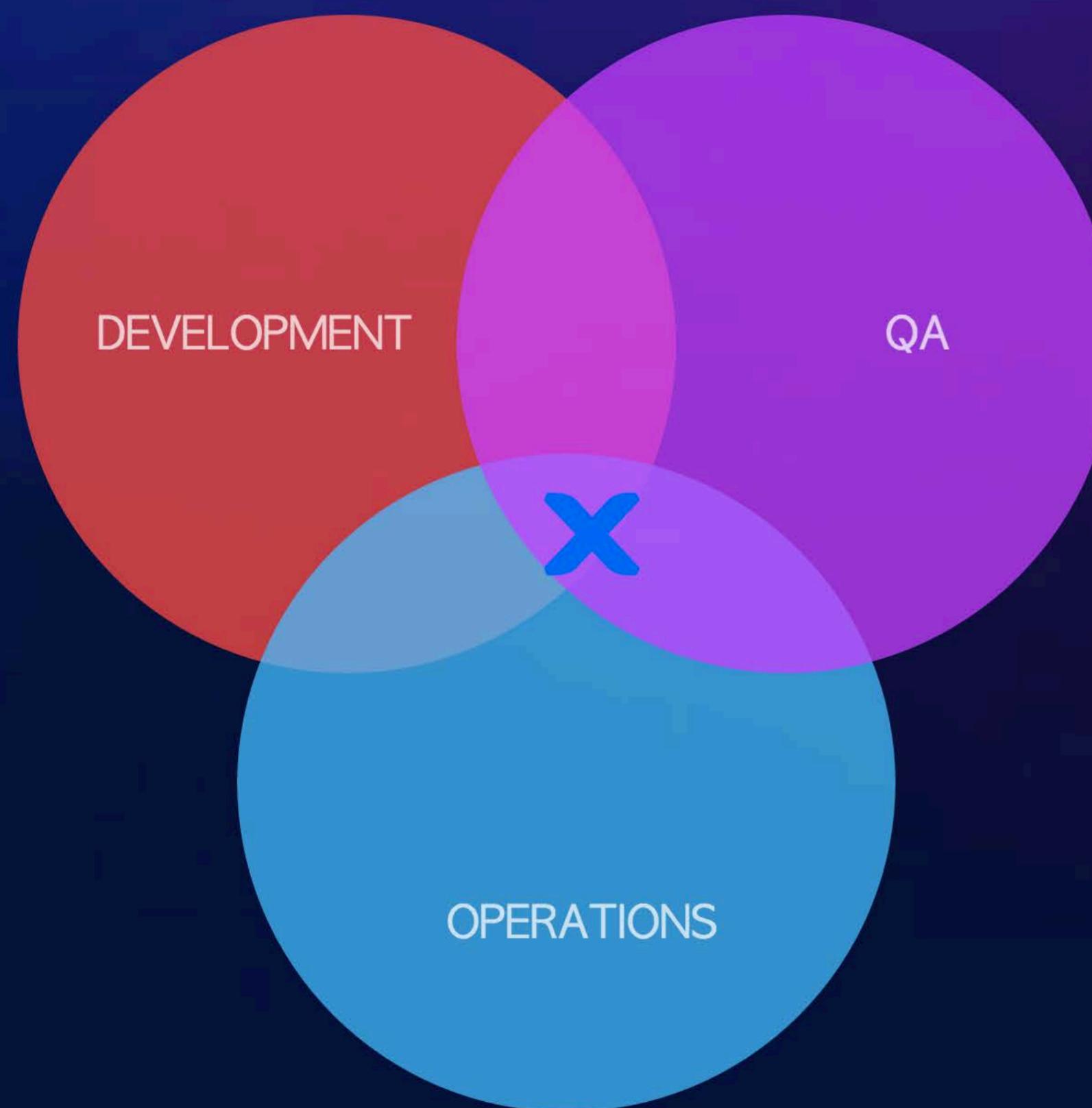
Associations - Theory

Concept



Associations - Theory

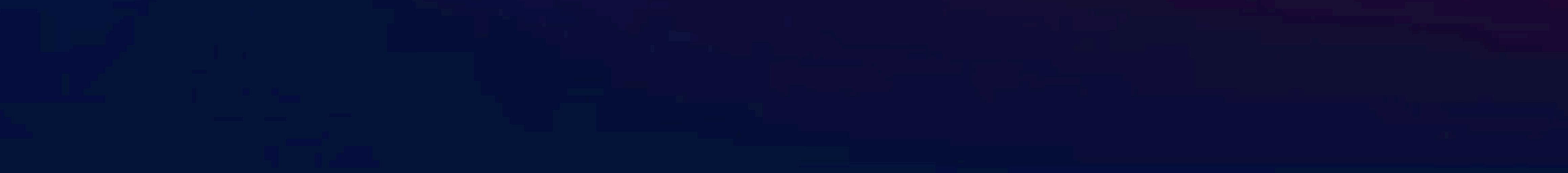
Concept



def.

defined as a “culture, movement or practice that emphasizes the collaboration and communication of software developers and other IT professionals”

Associations - Practice



Associations - Practice

Company

- Delivery Methodology ?
- Lifecycles ?
- Stake Owners ?

Associations - Practice

Company

- Delivery Methodology ?
- Lifecycles ?
- Stake Owners ?

Toolchain(s)

- Release
- Configuration
- Orchestration
- Monitoring
- Virtualization
- ...ization



Associations - Practice

Company

- Delivery Methodology ?
- Lifecycles ?
- Stake Owners ?

Toolchain(s)

- Release
- Configuration
- Orchestration
- Monitoring
- Virtualization
- ...ization

Policies

- Step 1 - ∞
- Products 1 - ?
- Have We? Can I? syndrome



Question

You Need it.
But What Really
Works?

A Simple Plan

```
if ( your-org == real-business )
then ( require ( 'devOps' ))
else ( fail + then require( 'devOpsLite' ));
```

DevOps.proc

- proc_1() - Big Picture vs. Reality
- proc_2() - Business, and By-Products
- proc_3() - Bring together, Break apart, then Re-assemble
- proc_4() - Start Smart, it Never Stops
- end() - Here's an example

proc_1() - Big Picture



Enterprise Scale Ecosystem

DevOps Must(s)

proc_1() - Big Picture



DevOps Must(s)

- Far Exceed Multi-Source User Expectation

proc_1() - Big Picture



Enterprise Scale Ecosystem

DevOps Must(s)

- Far Exceed Multi-Source User Expectation
- Be Integrated and Automated

proc_1() - Big Picture



Enterprise Scale Ecosystem

DevOps Must(s)

- Far Exceed Multi-Source User Expectation
- Be Integrated and Automated
- Be Simple, Secure and FTE

proc_1() - Big Picture



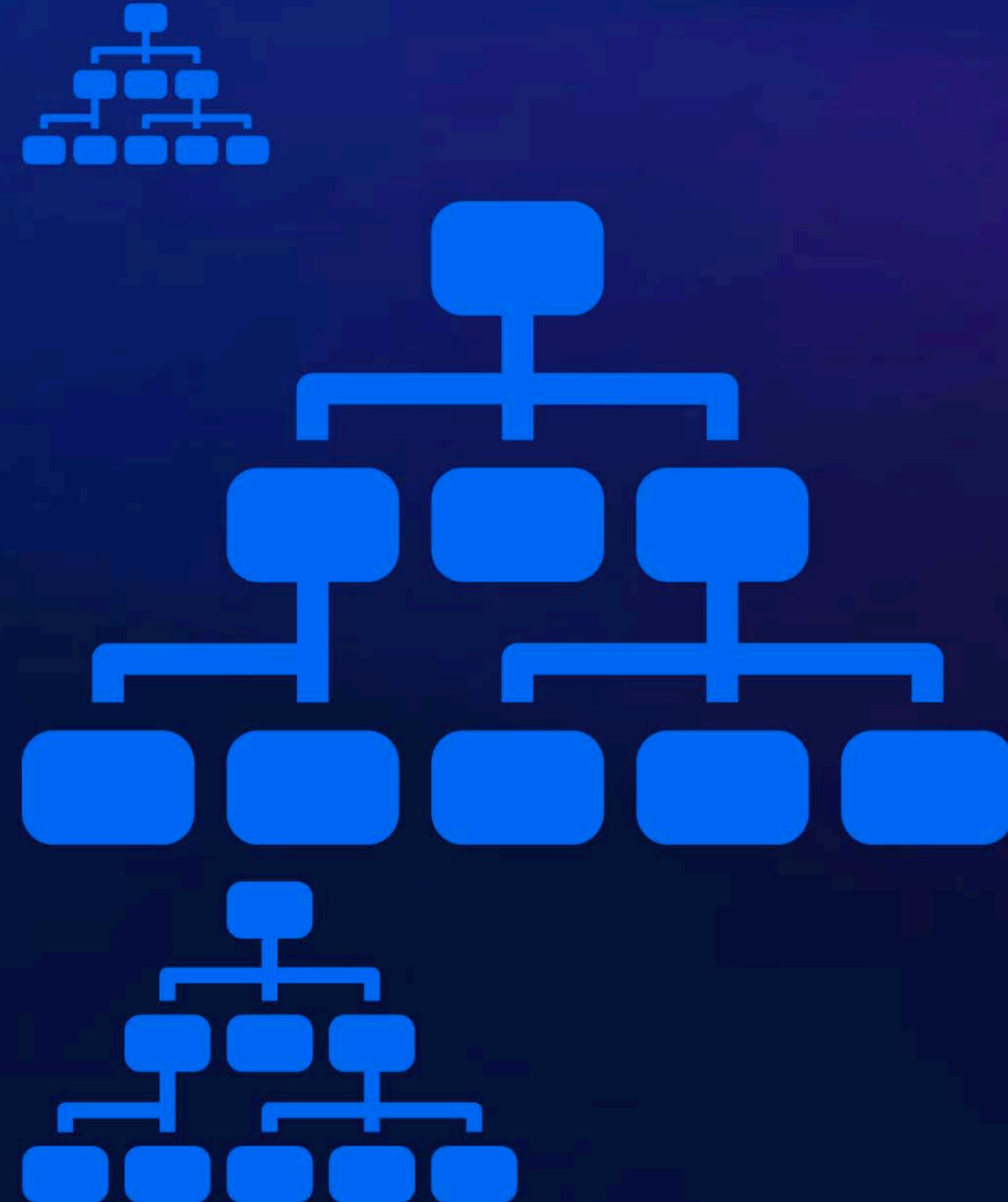
Enterprise Scale Ecosystem

DevOps Must(s)

- Far Exceed Multi-Source User Expectation
- Be Integrated and Automated
- Be Simple, Secure and FTE
- Increase Revenue(s) and Free

proc_1() - Realistic Picture

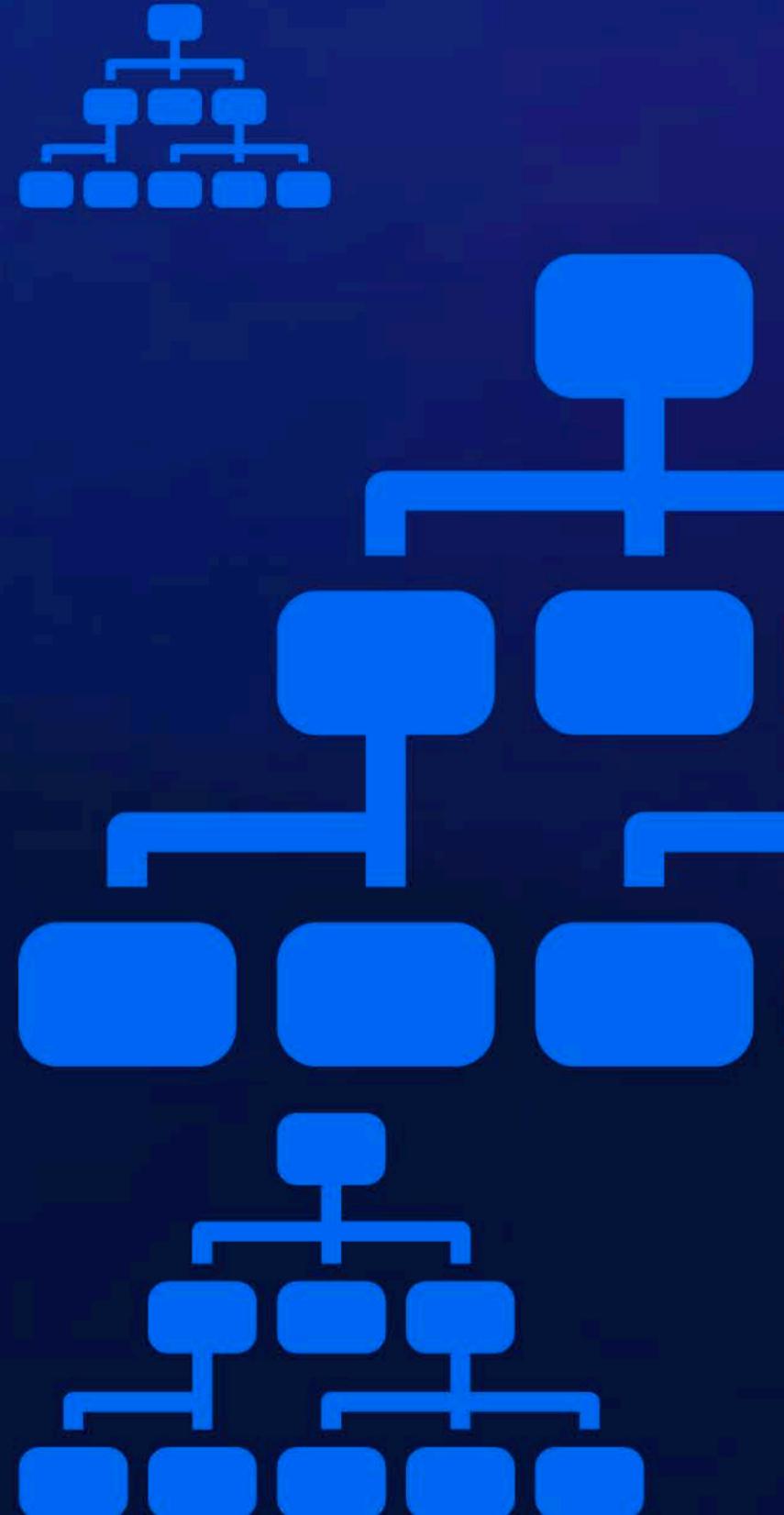
Most Organizations



proc_1() - Realistic Picture

Most Organizations

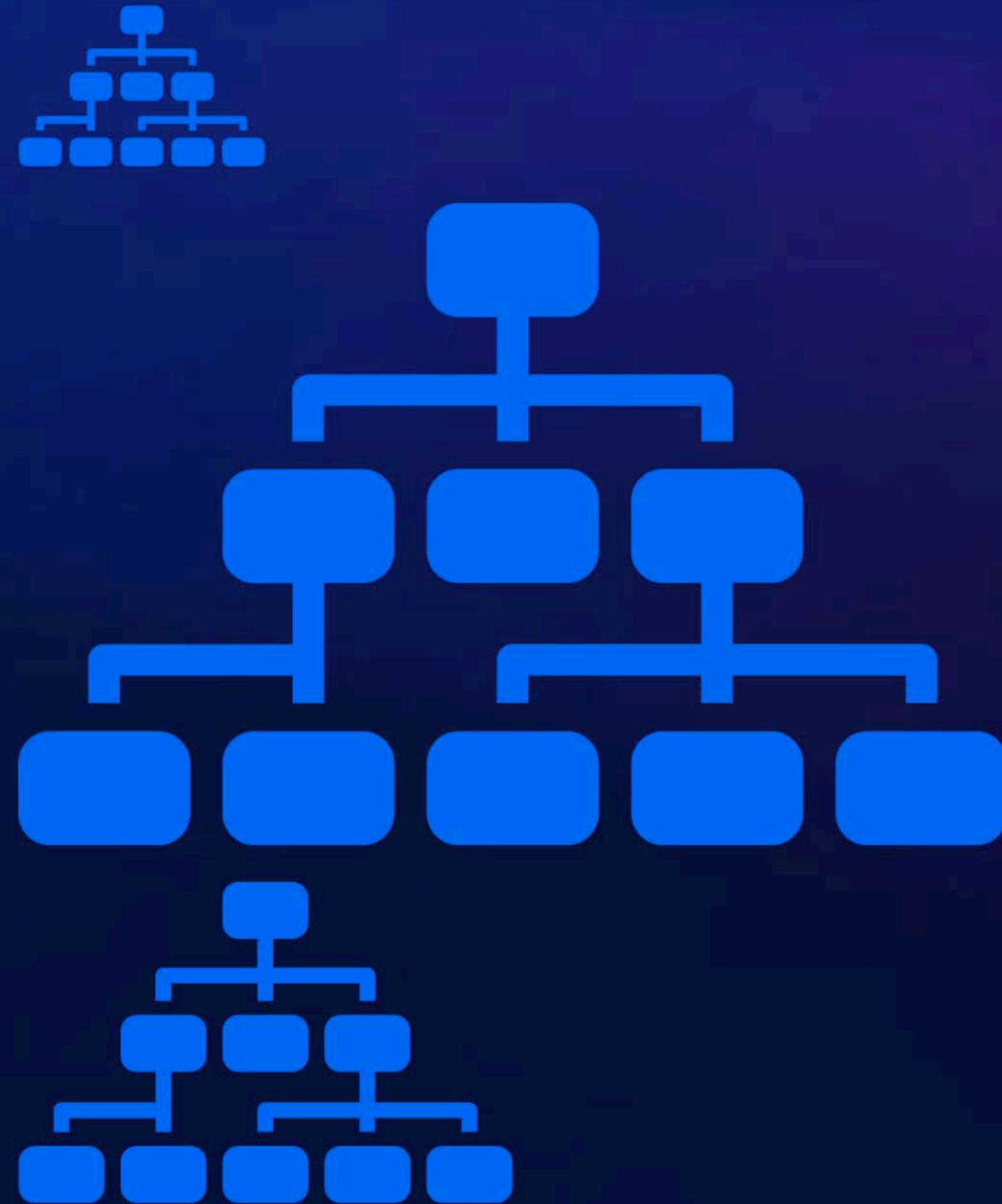
- Immediate ROI Focused



proc_1() - Realistic Picture

Most Organizations

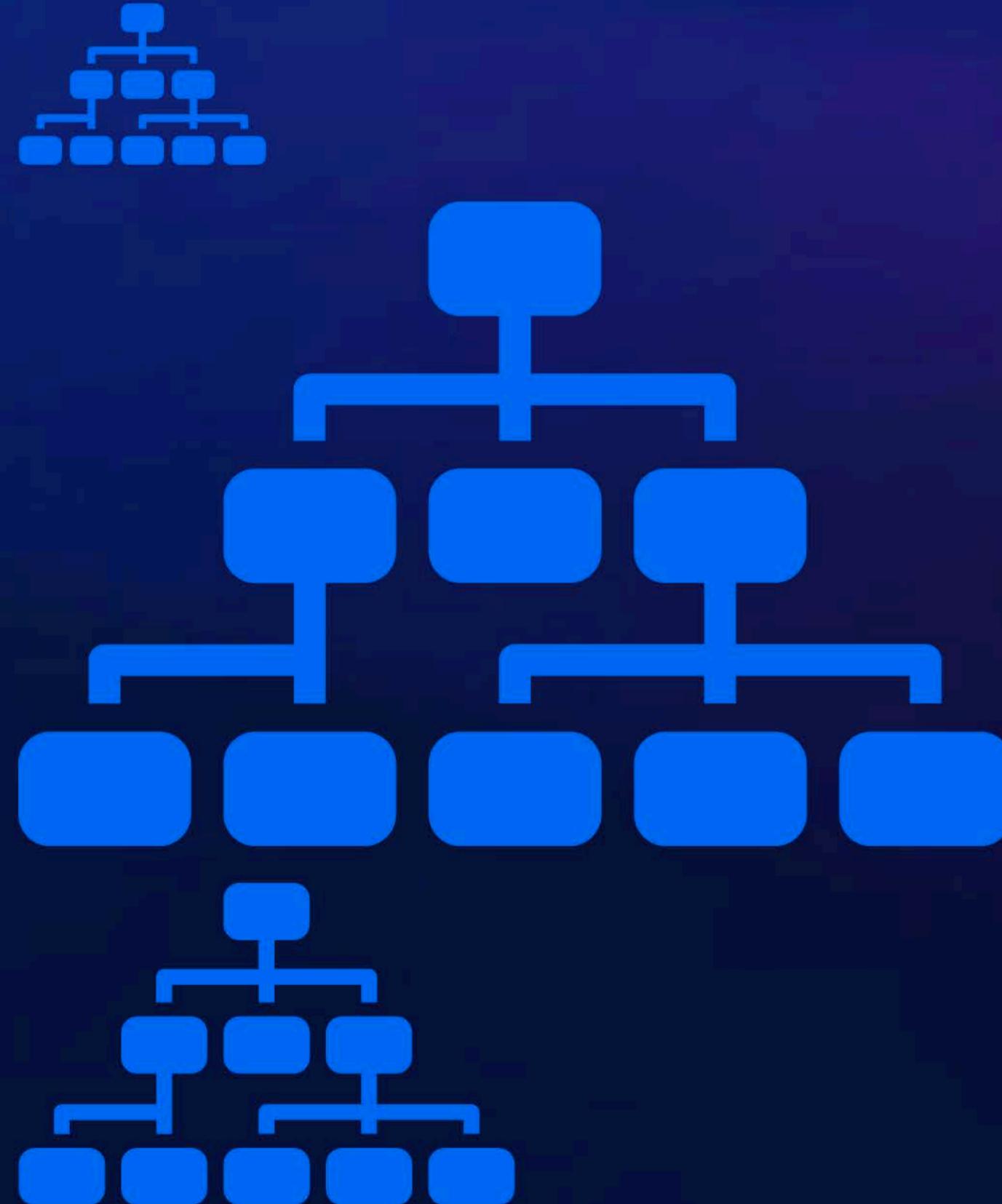
- Immediate ROI Focused
- Right Staffed or Just Collaborative



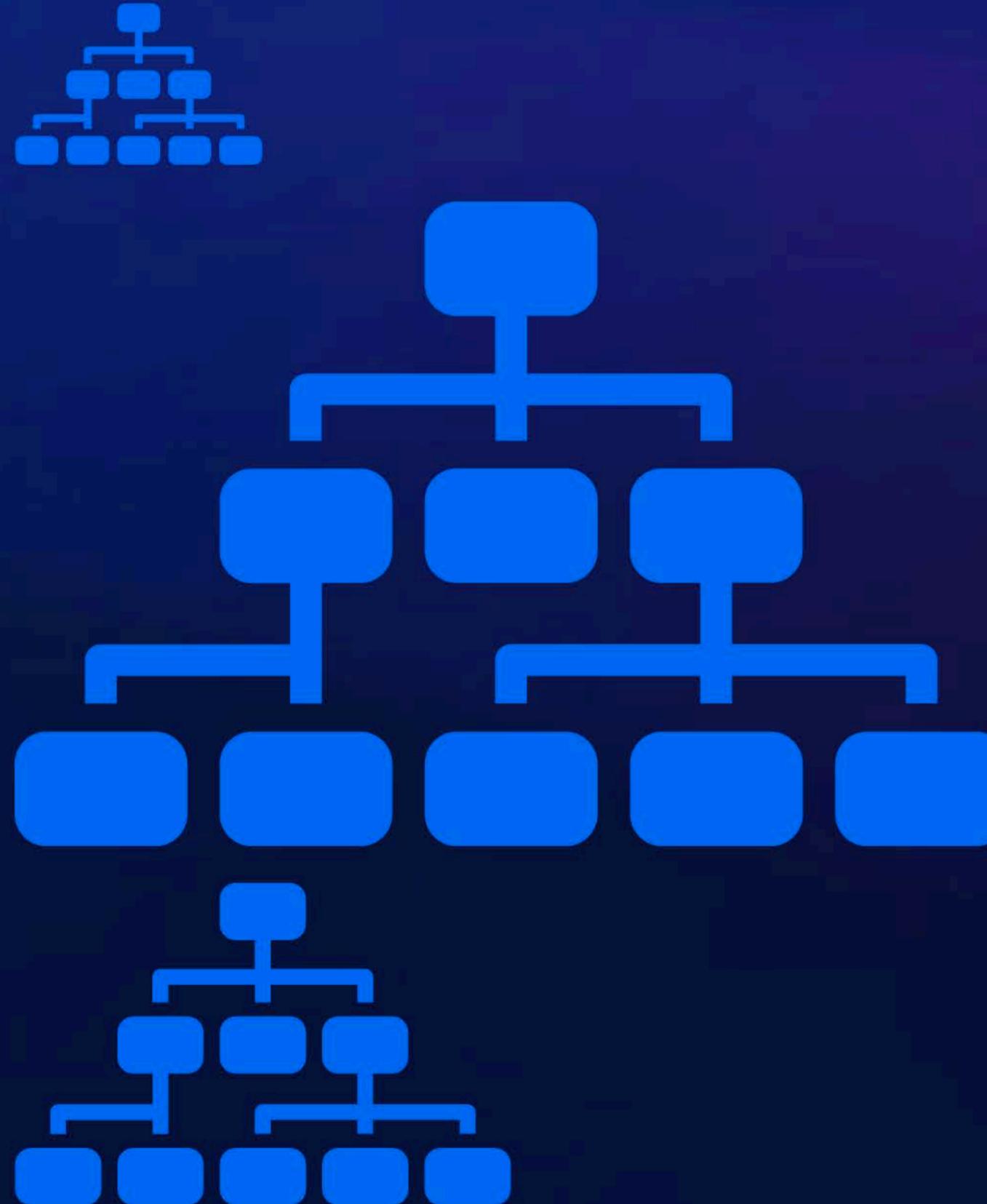
proc_1() - Realistic Picture

Most Organizations

- Immediate ROI Focused
- Right Staffed or Just Collaborative
- Want World Record without Exercising



proc_1() - Realistic Picture



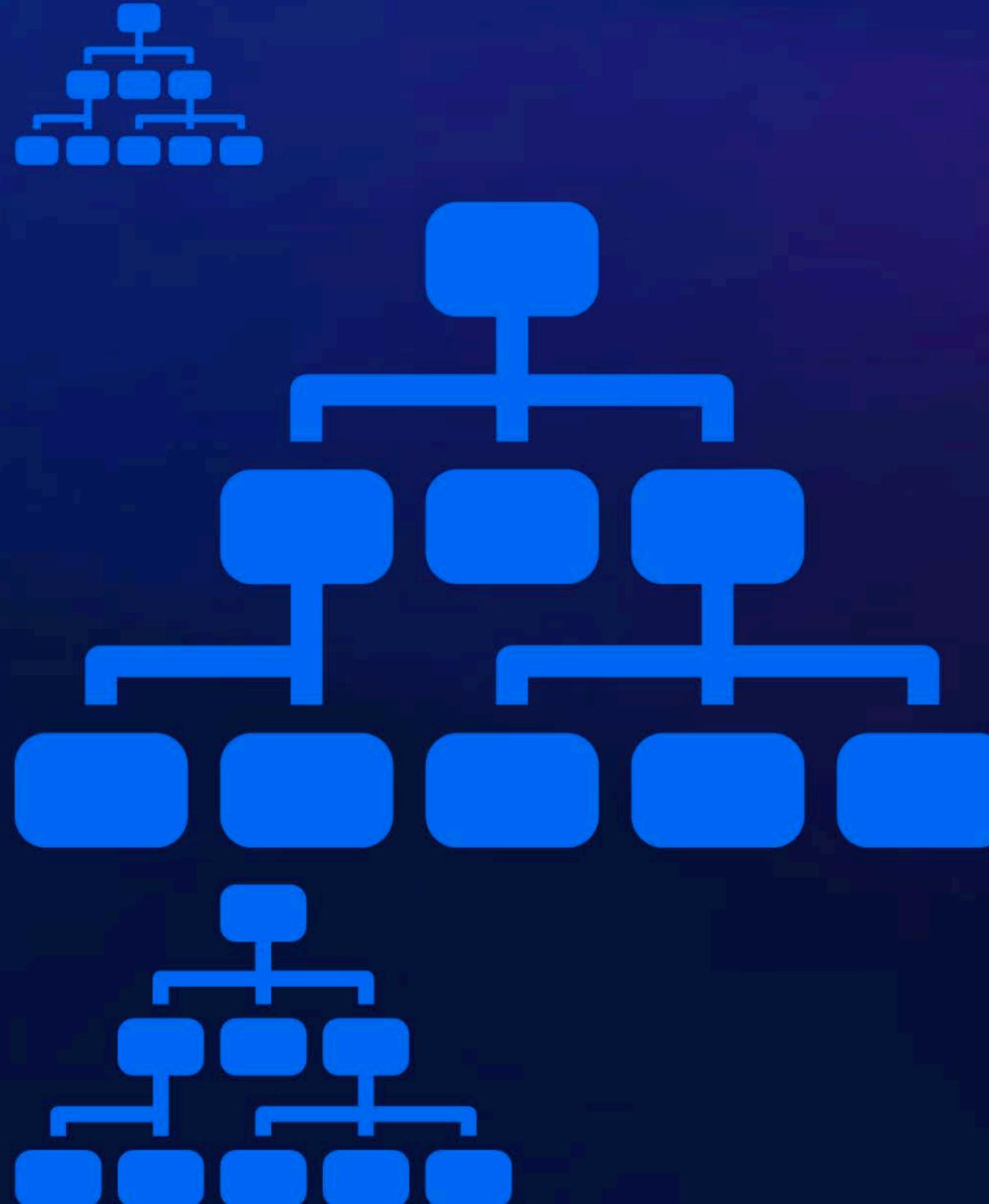
Most Organizations

- Immediate ROI Focused
- Right Staffed or Just Collaborative
- Want World Record without Exercising

DevOps Must(s)

- Developer Lead Operations

proc_1() - Realistic Picture



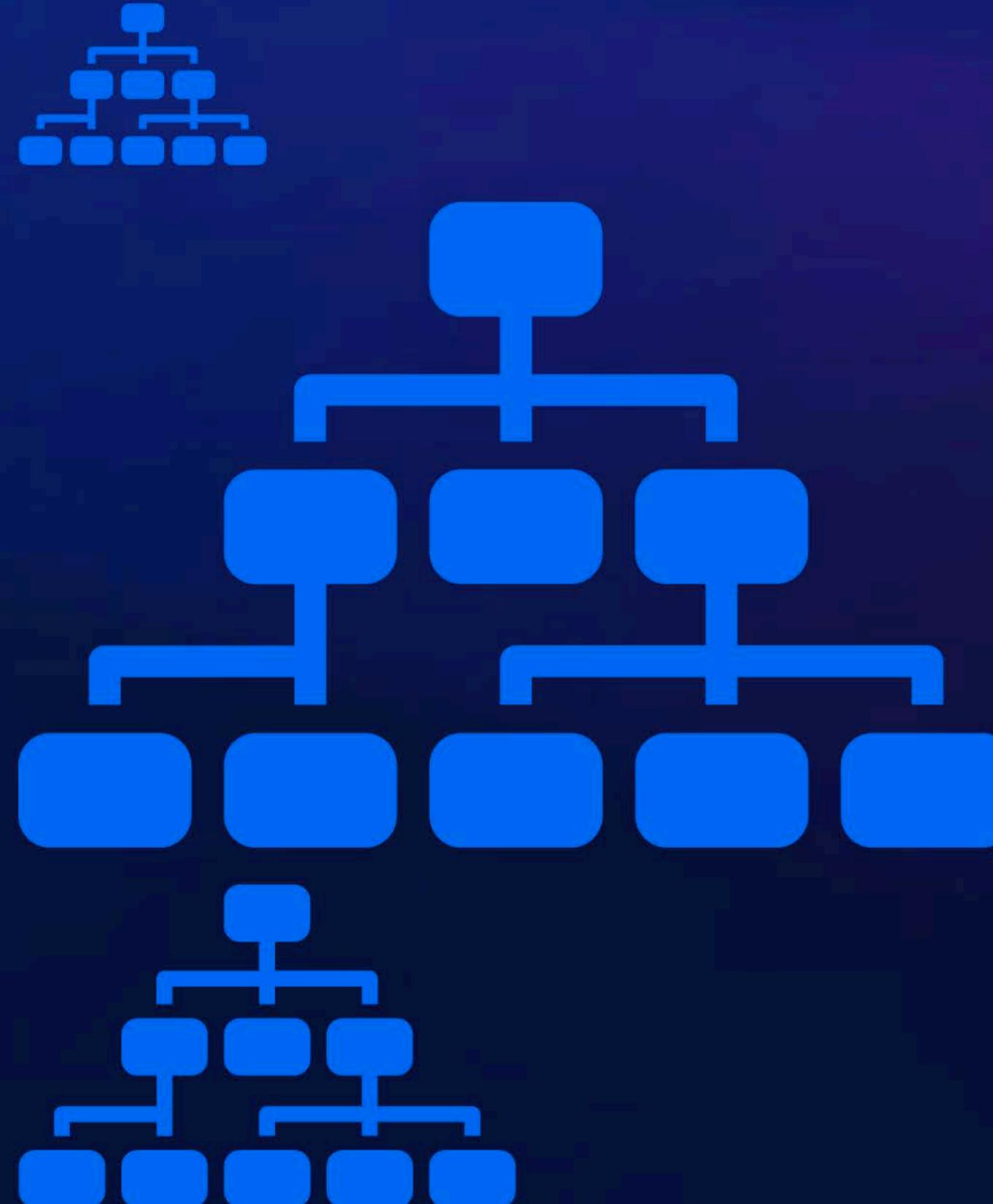
Most Organizations

- Immediate ROI Focused
- Right Staffed or Just Collaborative
- Want World Record without Exercising

DevOps Must(s)

- Developer Lead Operations
- Be Next-Generation Viral

proc_1() - Realistic Picture



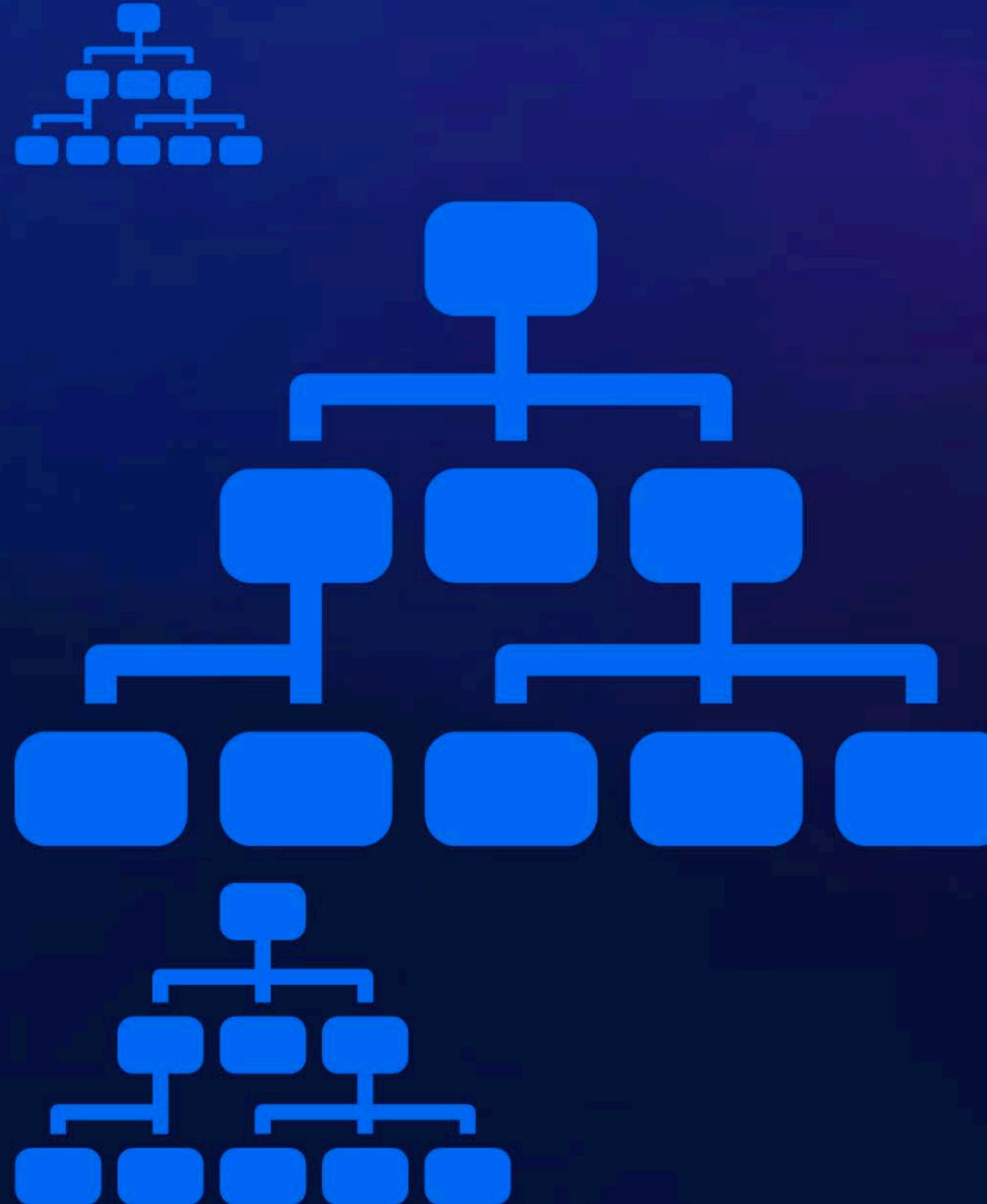
Most Organizations

- Immediate ROI Focused
- Right Staffed or Just Collaborative
- Want World Record without Exercising

DevOps Must(s)

- Developer Lead Operations
- Be Next-Generation Viral
- Not get in the way

proc_1() - Realistic Picture



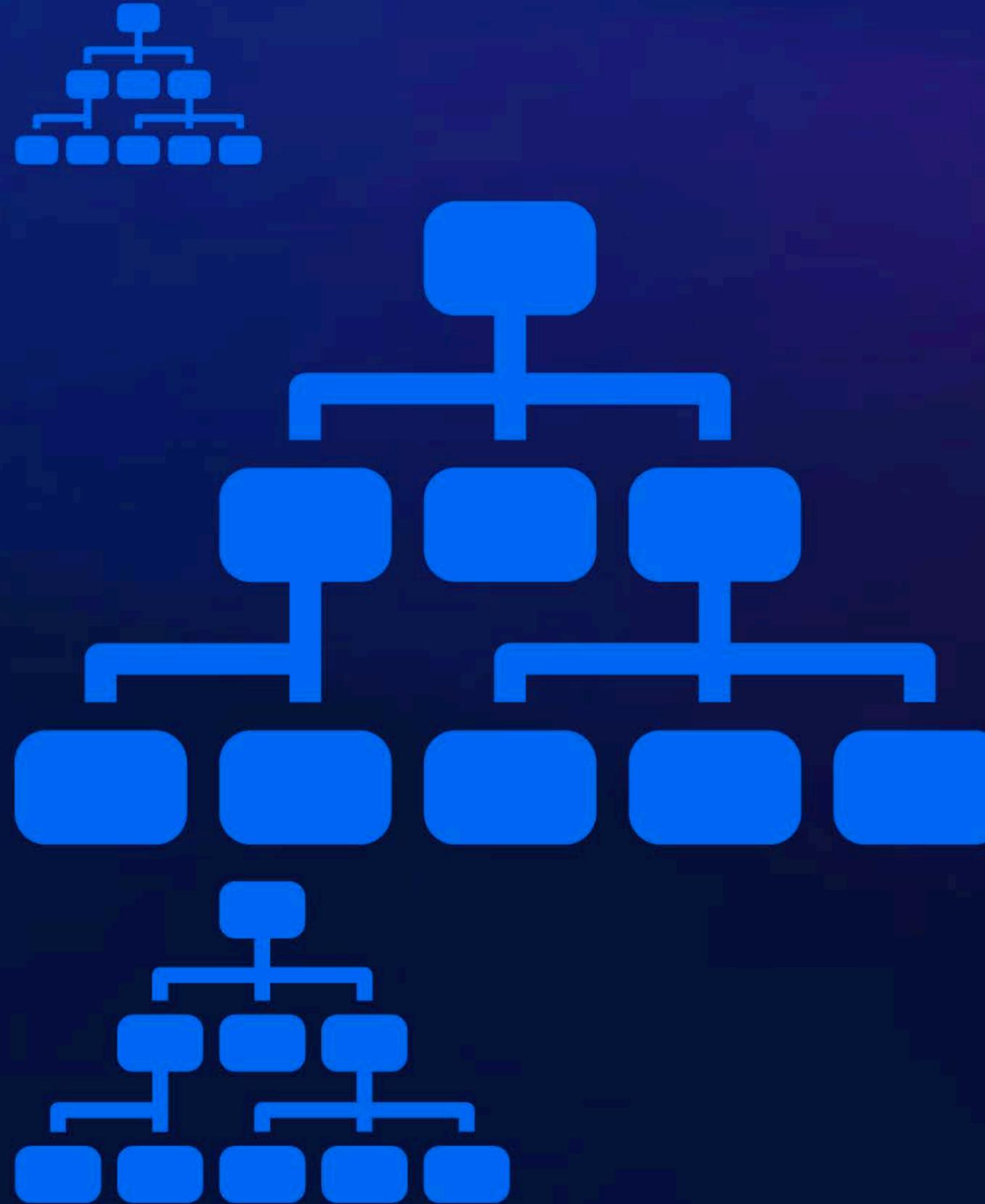
Most Organizations

- Immediate ROI Focused
- Right Staffed or Just Collaborative
- Want World Record without Exercising

DevOps Must(s)

- Developer Lead Operations
- Be Next-Generation Viral
- Not get in the way
- Have Strict Controls

proc_1() - Realistic Picture



Most Organizations

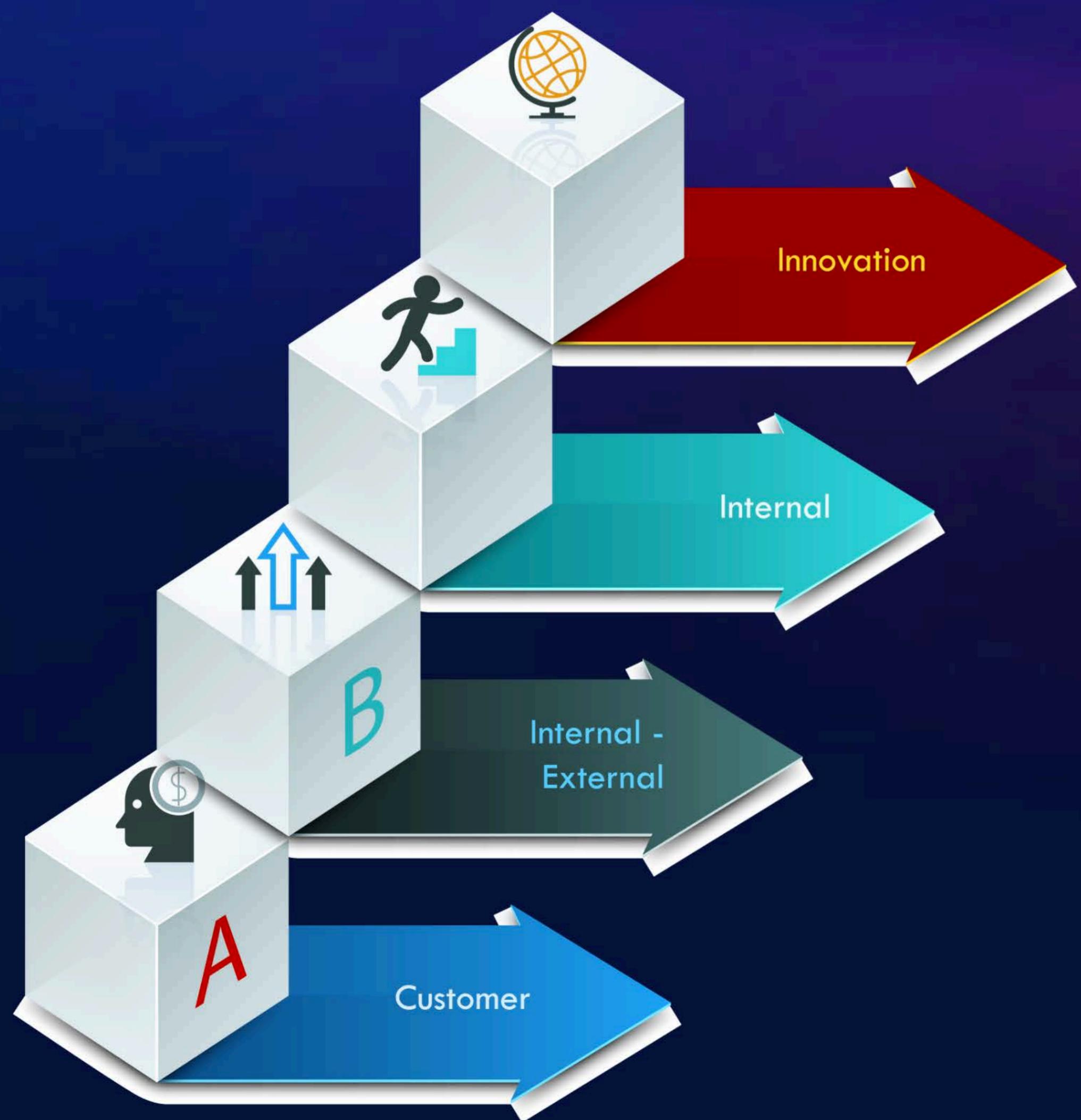
- Immediate ROI Focused
- Right Staffed or Just Collaborative
- Want World Record without Exercising

DevOps Must(s)

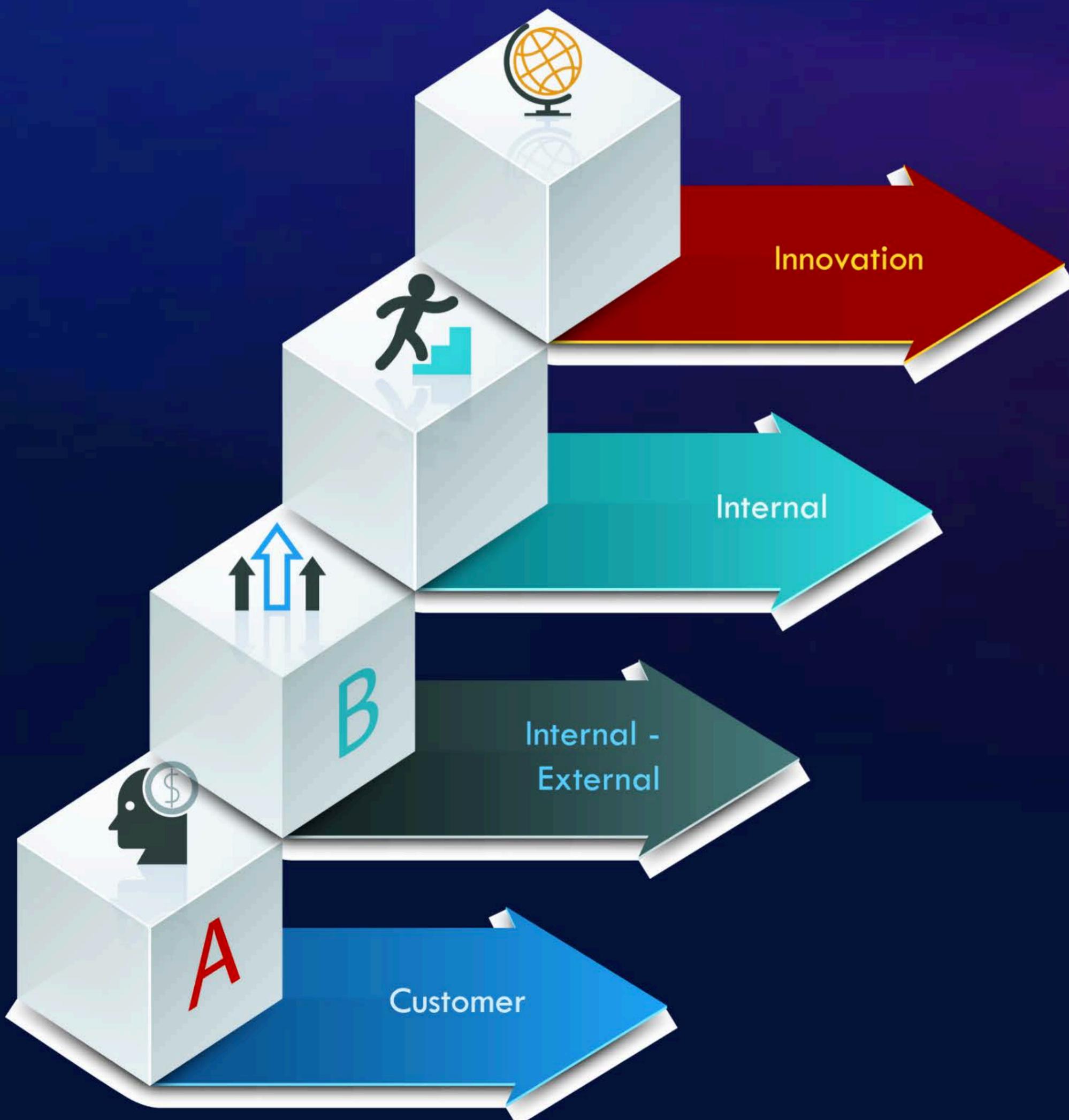
- Developer Lead Operations
- Be Next-Generation Viral
- Not get in the way
- Have Strict Controls
- Encourage participation through shaming

proc_2() - Your Business and By-Products

By Products



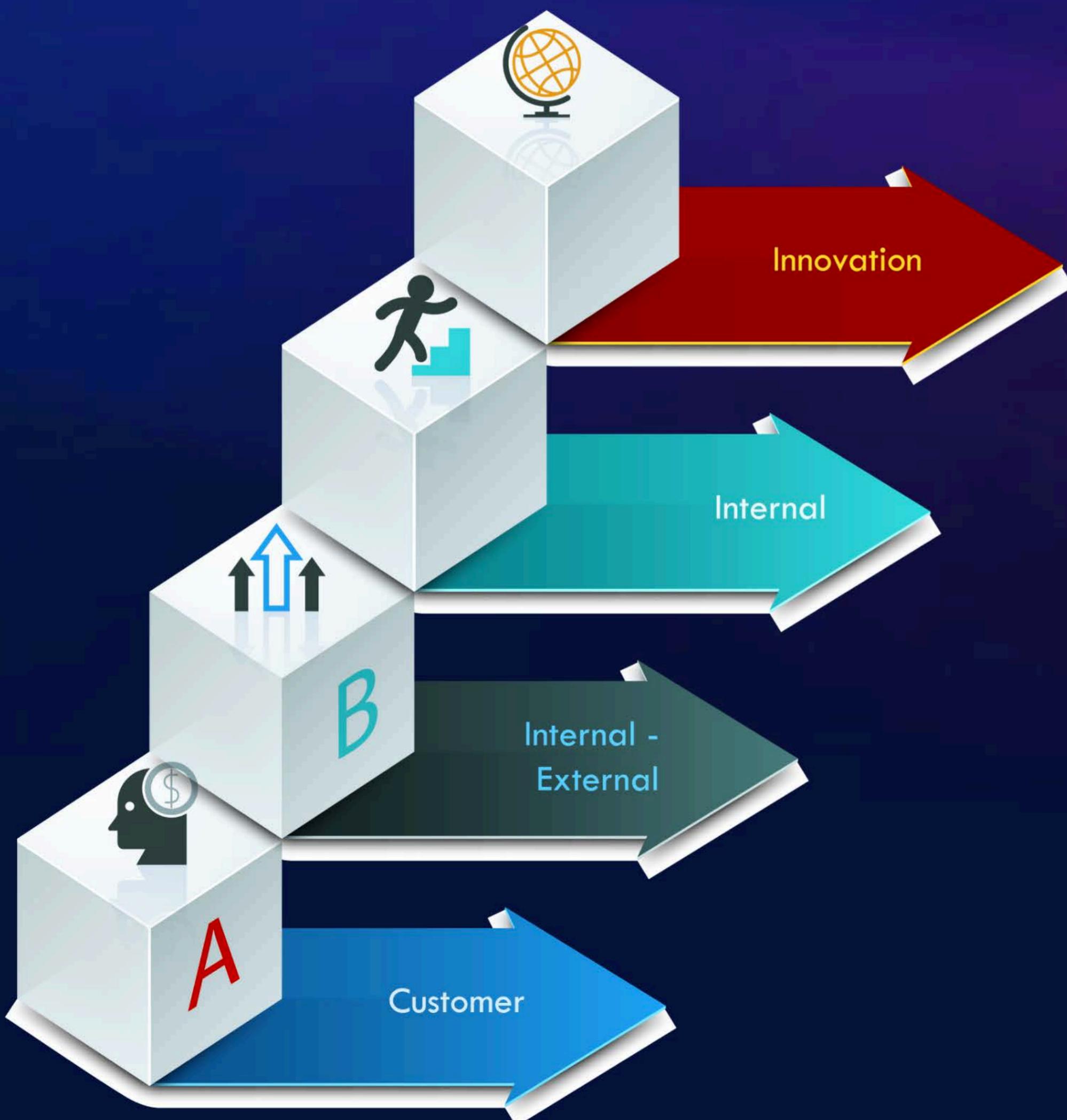
proc_2() - Your Business and By-Products



By Products

- Micro Apps the new Proto-types

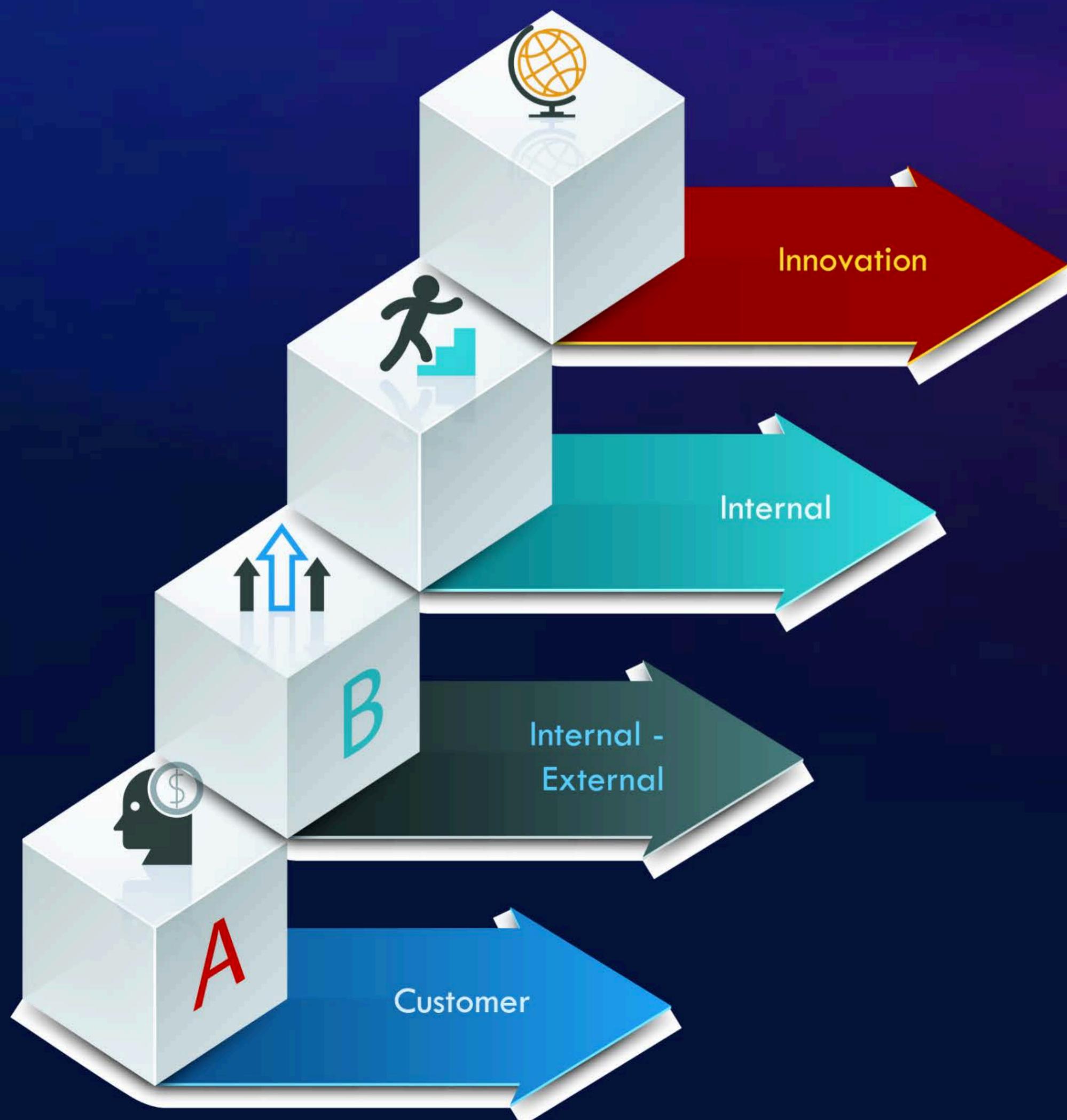
proc_2() - Your Business and By-Products



By Products

- Micro Apps the new Prodo-types
- Internal projects are unorganized

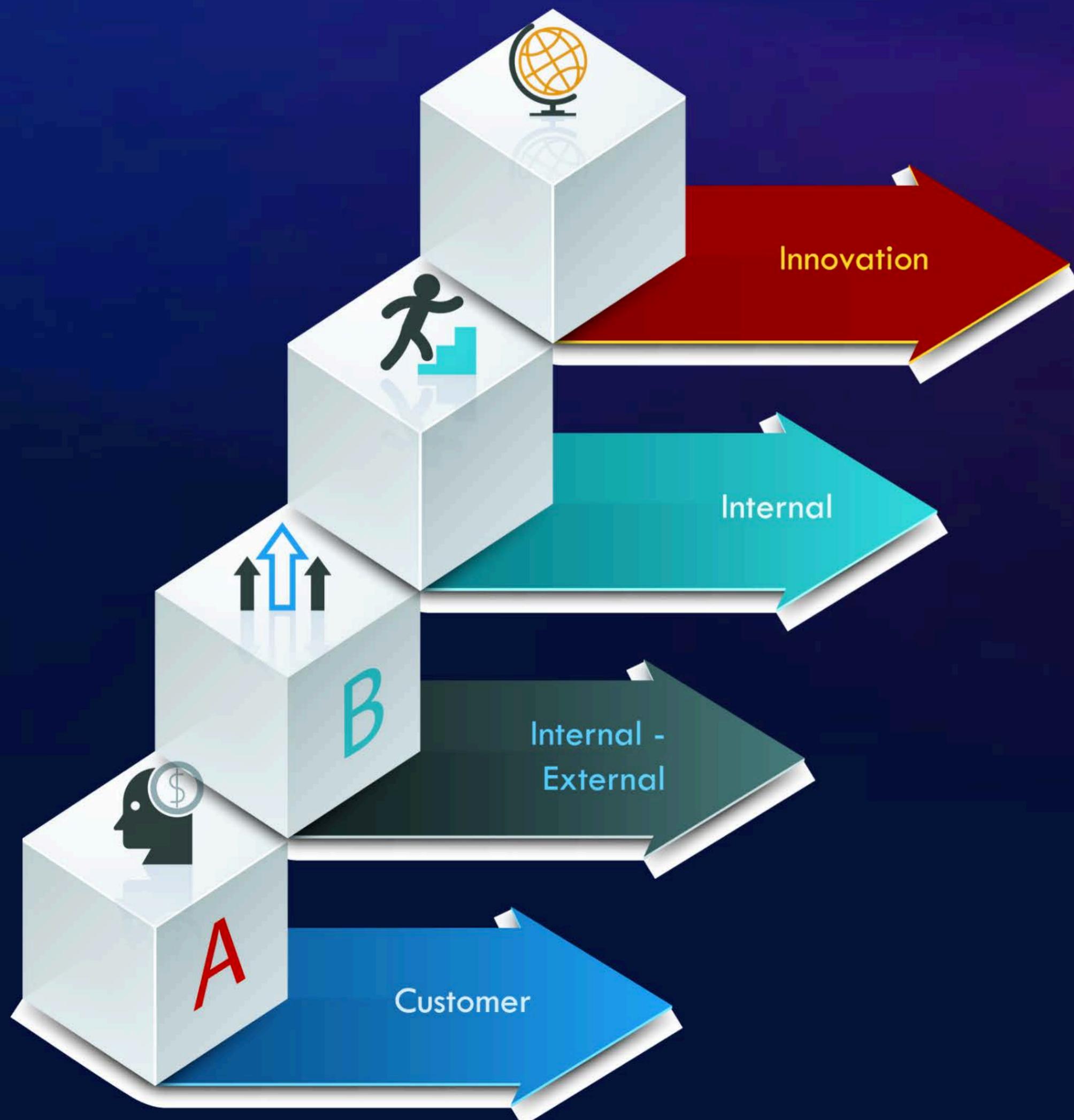
proc_2() - Your Business and By-Products



By Products

- Micro Apps the new Prodo-types
- Internal projects are unorganized
- Connected is complex

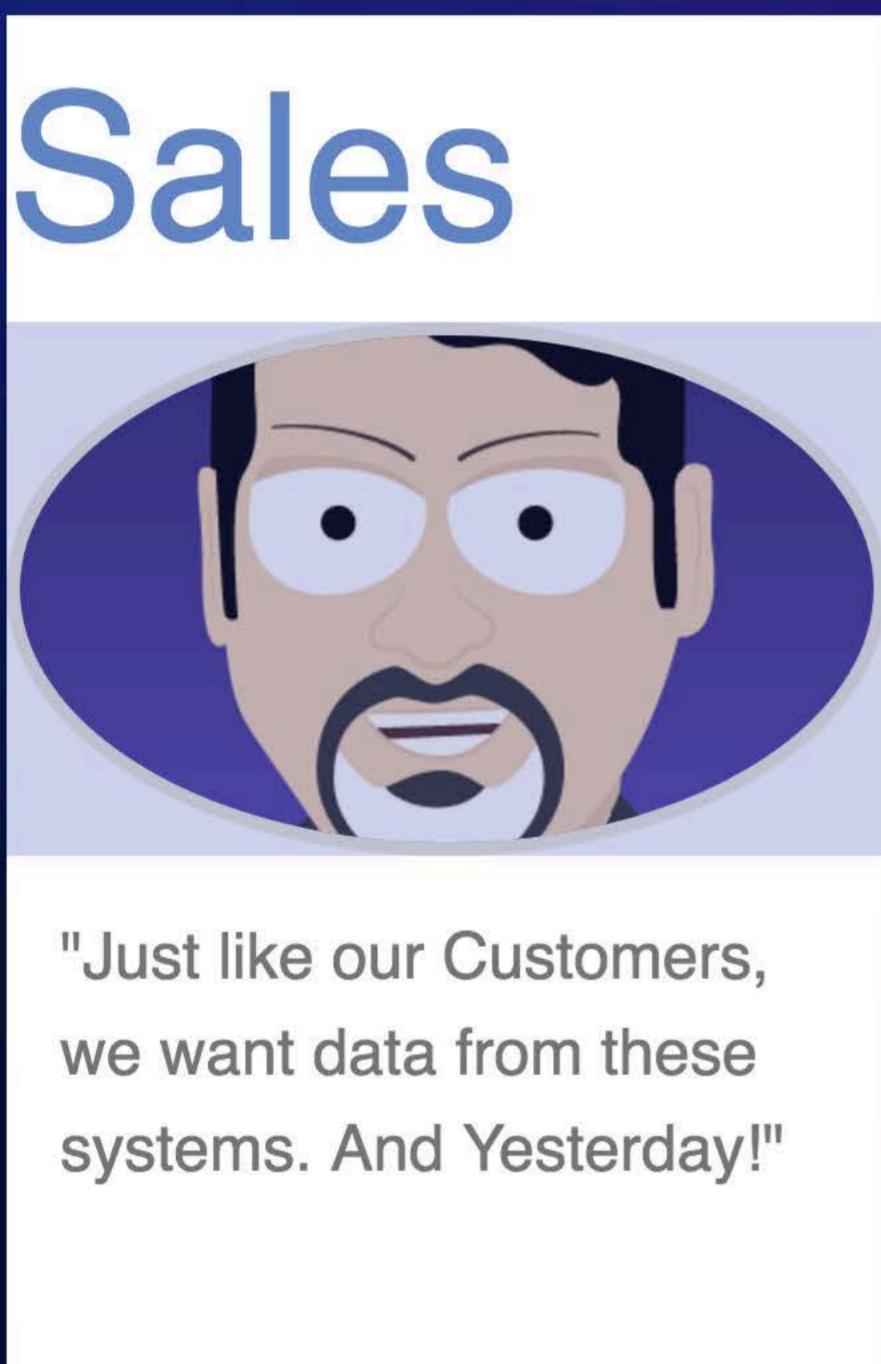
proc_2() - Your Business and By-Products



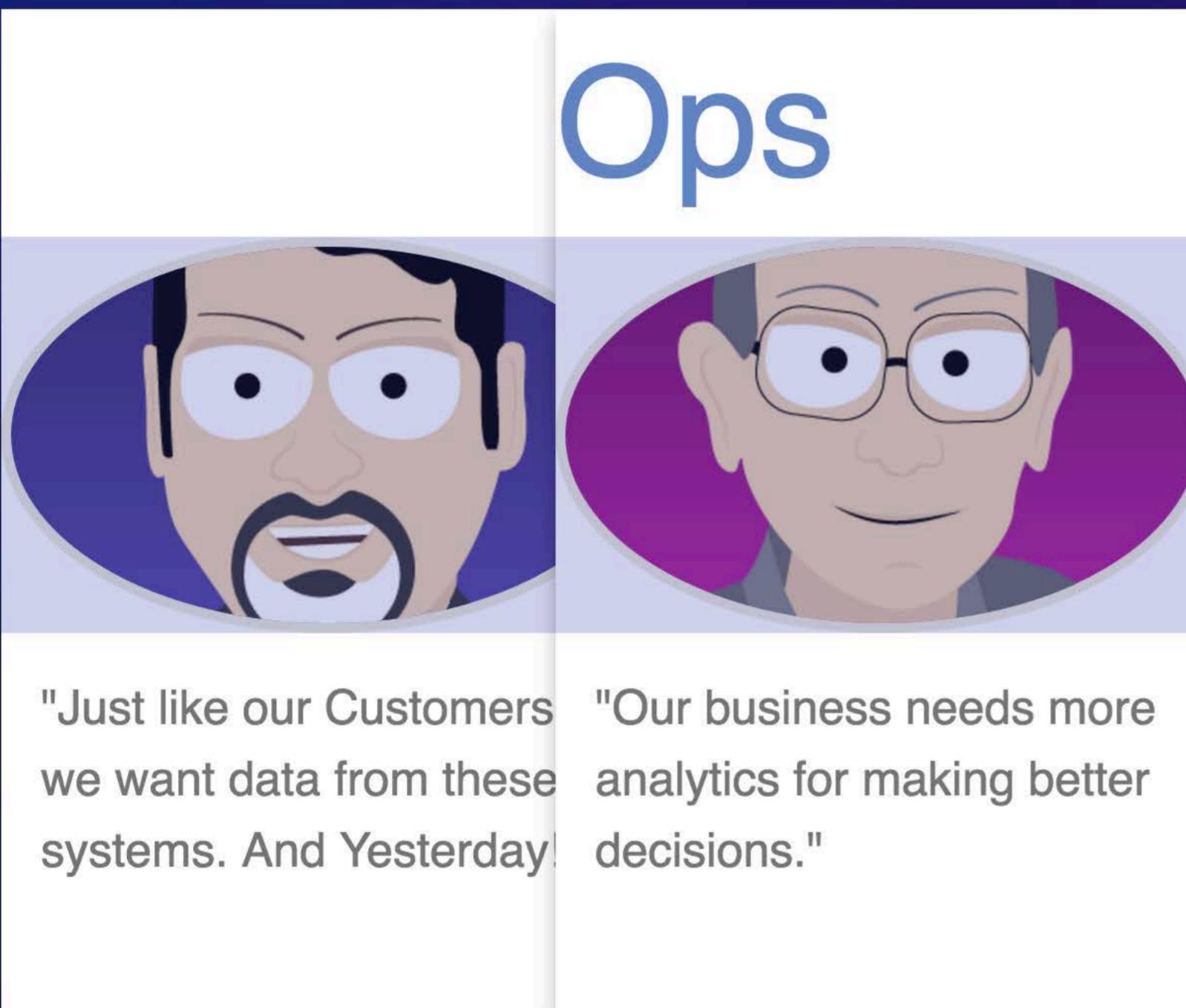
By Products

- Micro Apps the new Prodo-types
- Internal projects are unorganized
- Connected is complex
- Custom one off

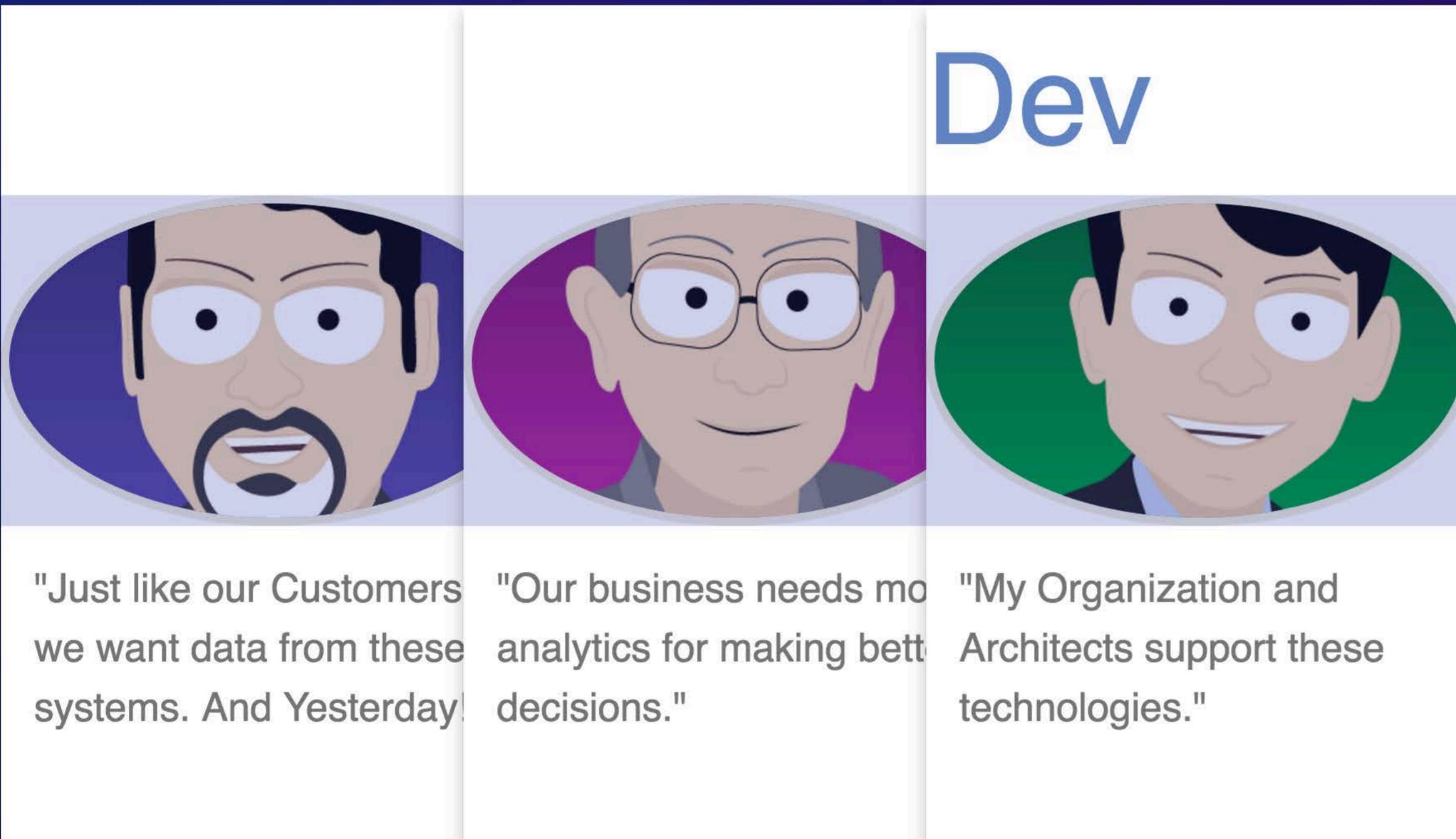
proc_3() - Bring together, Break apart



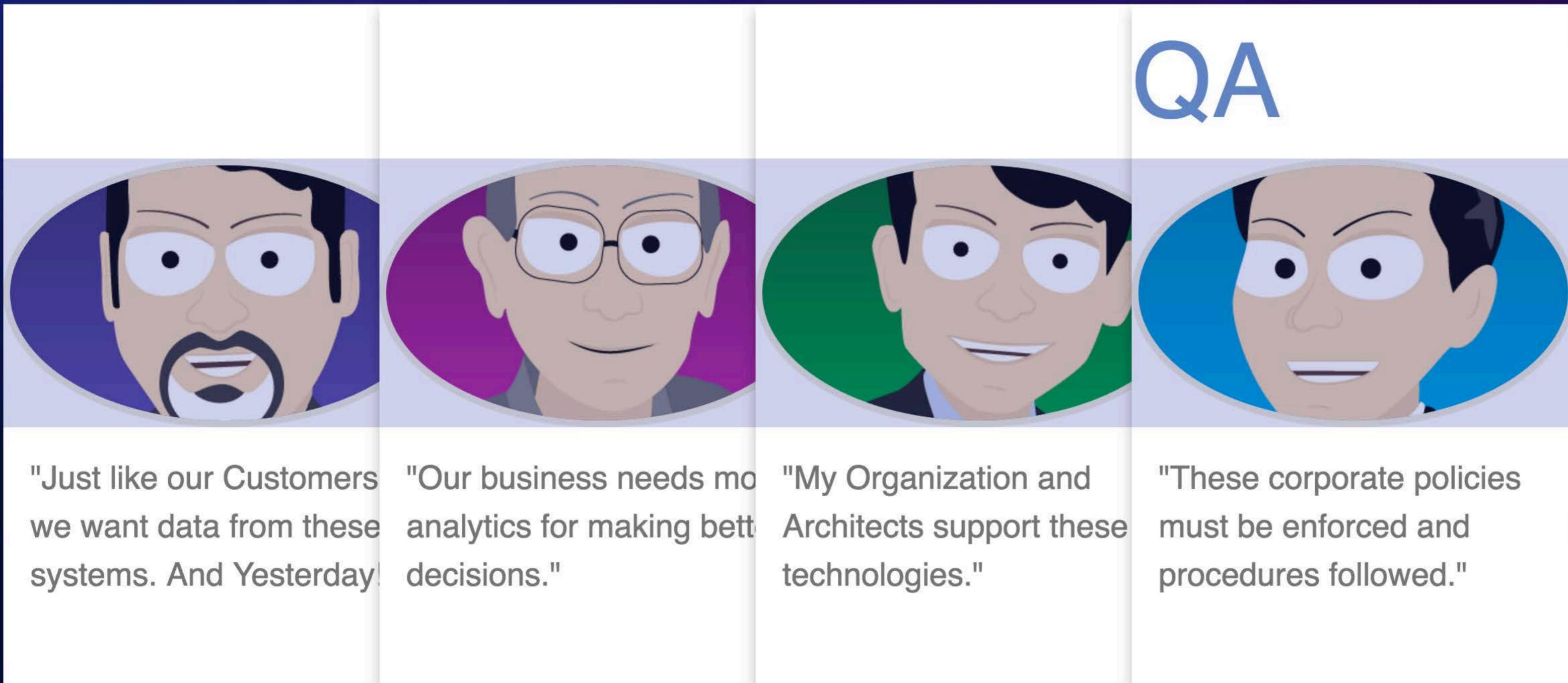
proc_3() - Bring together, Break apart



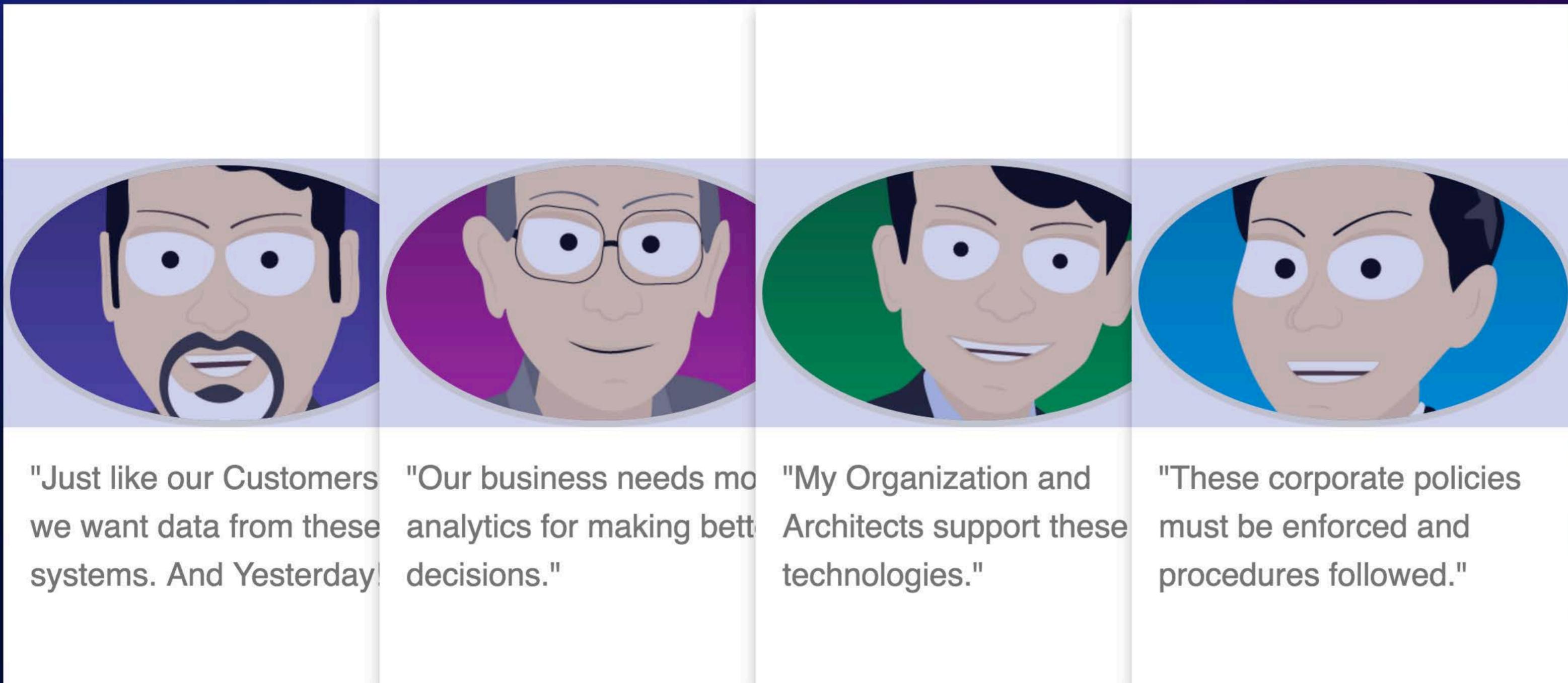
proc_3() - Bring together, Break apart



proc_3() - Bring together, Break apart



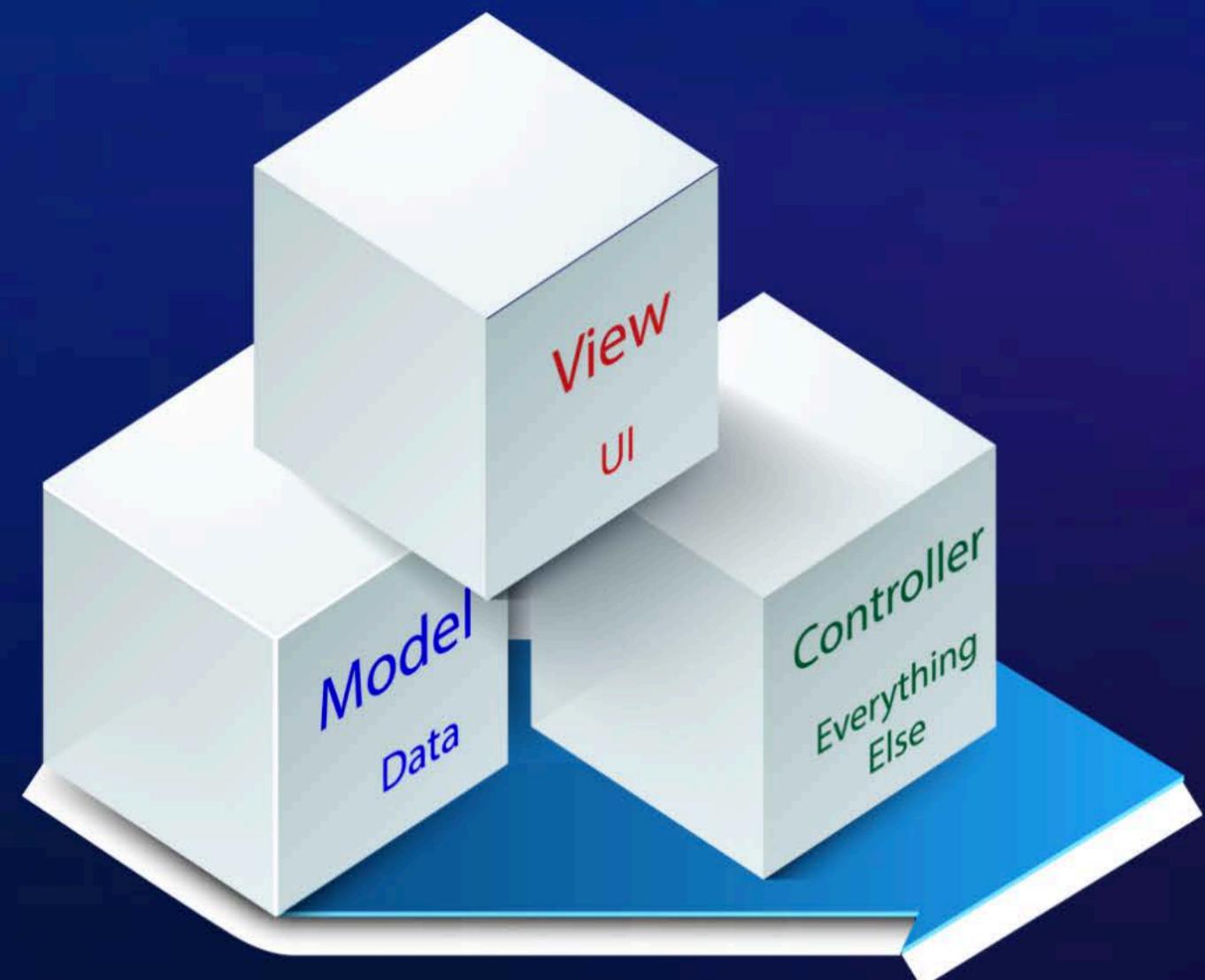
proc_3() - Bring together, Break apart



"They said that last time. Let's re-purpose the old stuff."

-Smart Person who understands DevOps

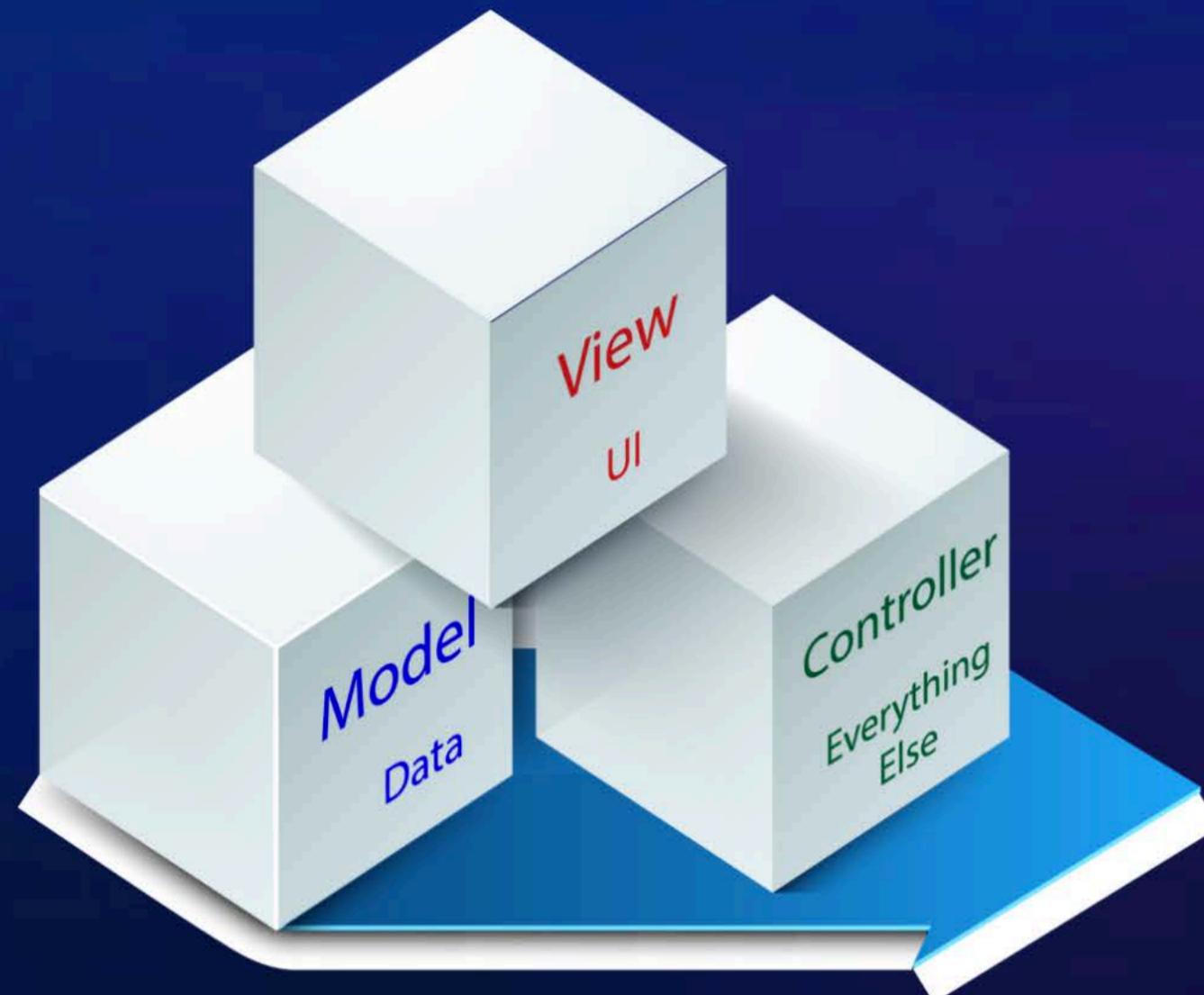
proc_3() - Re-assemble



Design Pattern of Everything

Design is key

proc_3() - Re-assemble

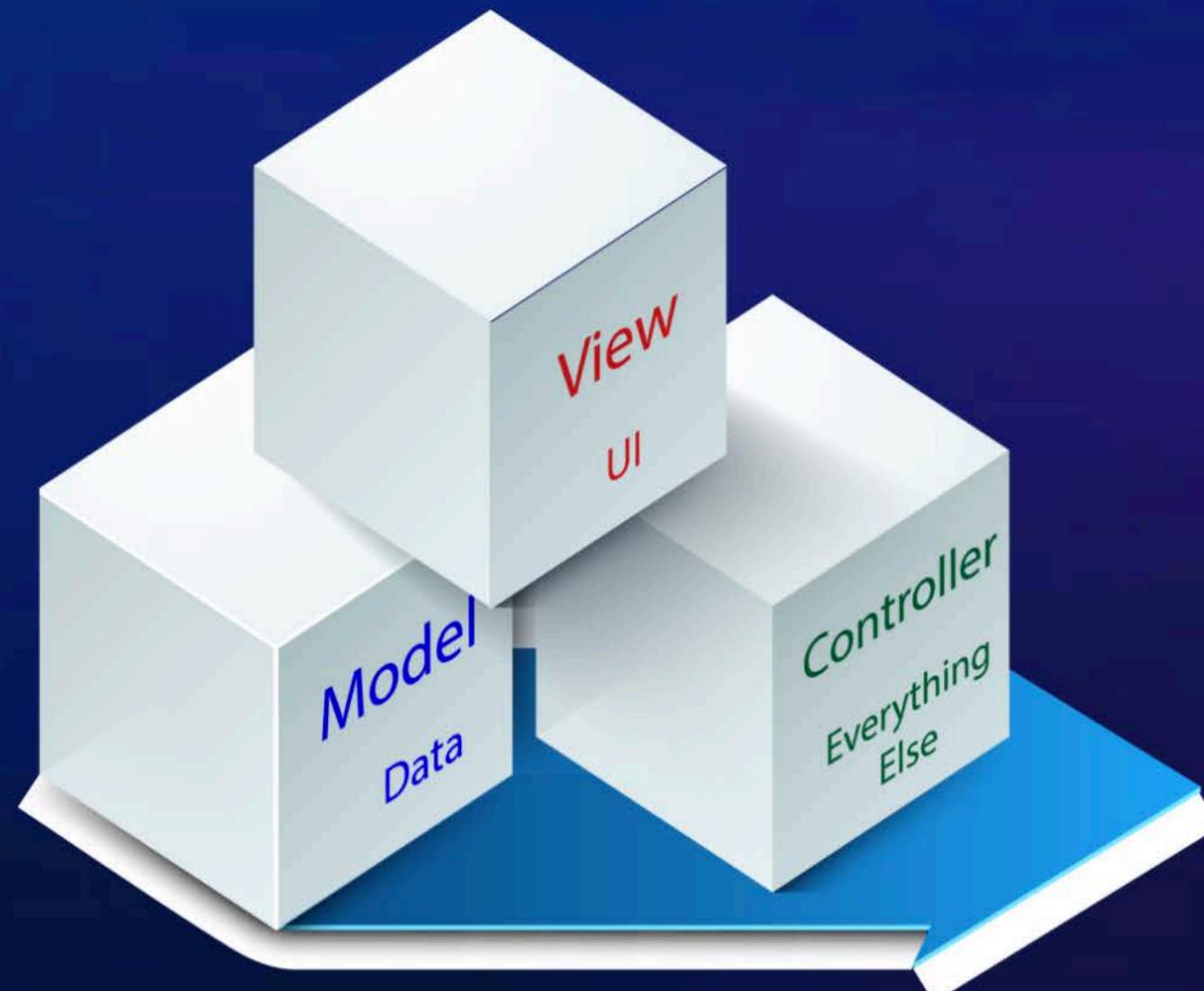


Design Pattern of Everything

Design is key

- Loosely coupled; tightly encapsulated; distributable; continuable

proc_3() - Re-assemble

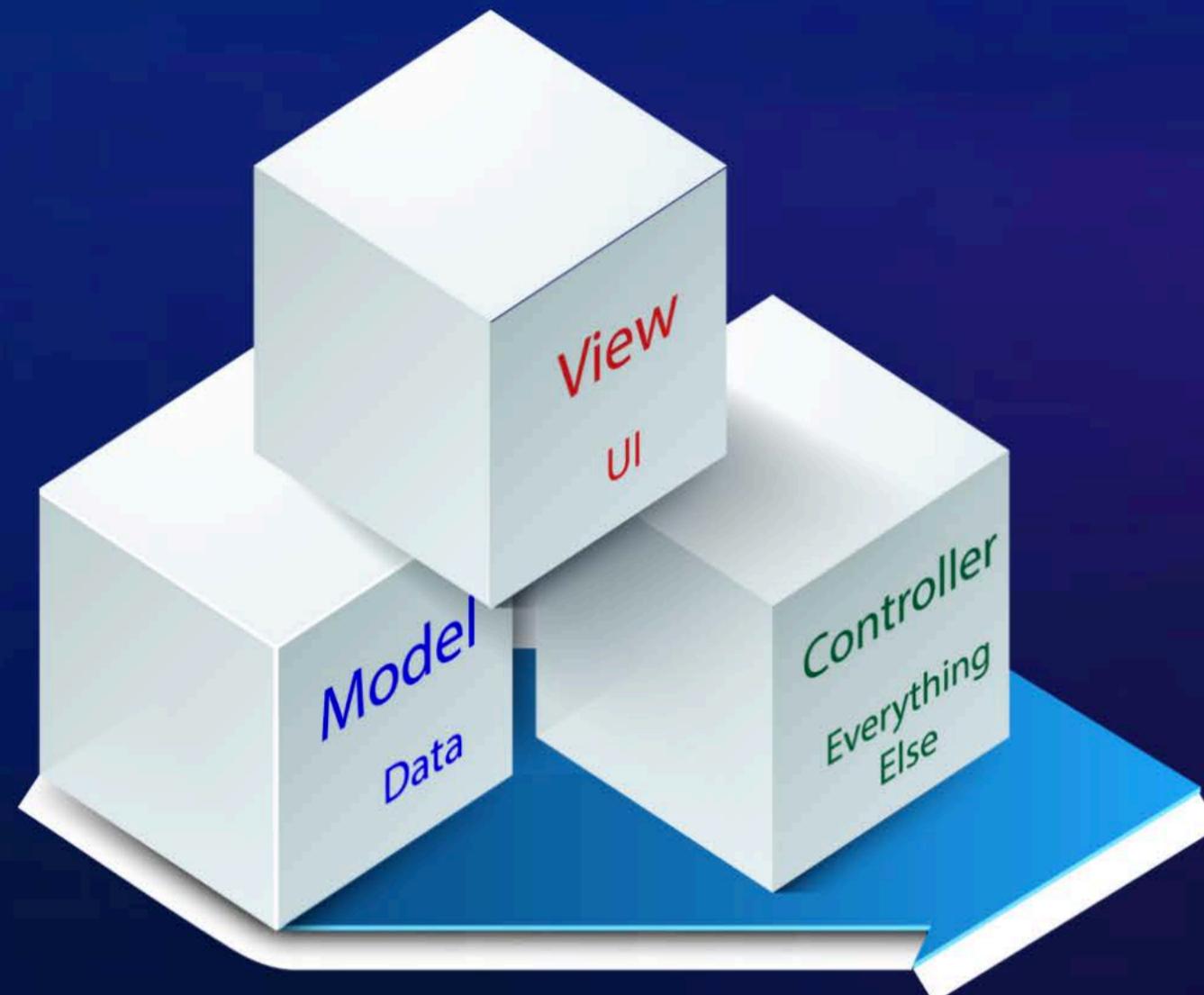


Design Pattern of Everything

Design is key

- Loosely coupled; tightly encapsulated; distributable; continuable
- Balance between in-progress and in production

proc_3() - Re-assemble

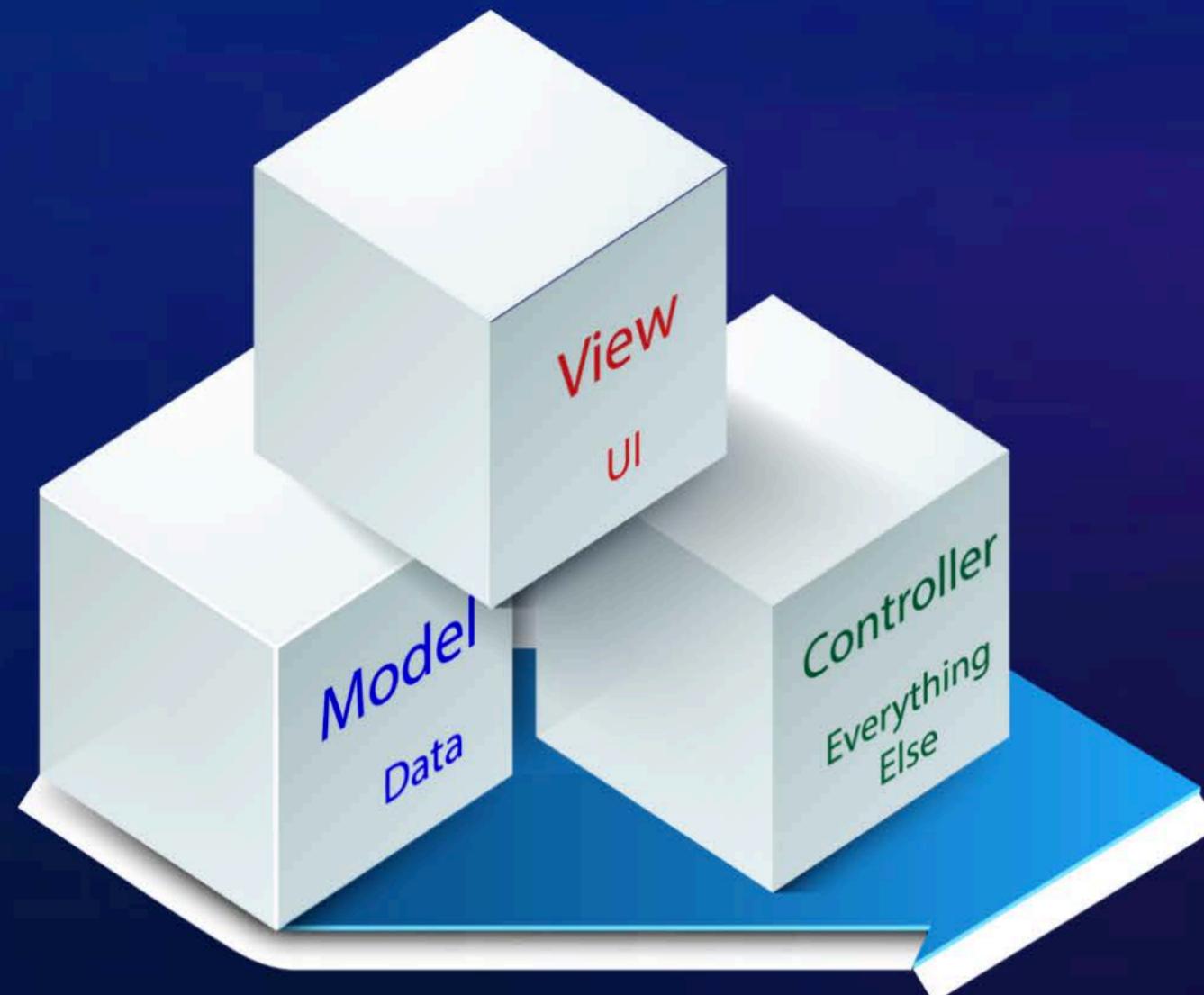


Design Pattern of Everything

Design is key

- Loosely coupled; tightly encapsulated; distributable; continuable
- Balance between in-progress and in production
- Common Practices Should Never Be Written, Maybe only once.

proc_3() - Re-assemble



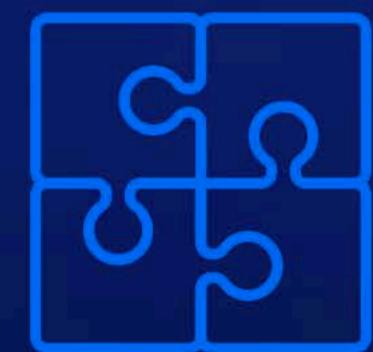
Design Pattern of Everything

Design is key

- Loosely coupled; tightly encapsulated; distributable; continuable
- Balance between in-progress and in production
- Common Practices Should Never Be Written, Maybe only once.
- Assemble and Integrate more than Engineer

proc_4() - Start Smart

Concepts across Company

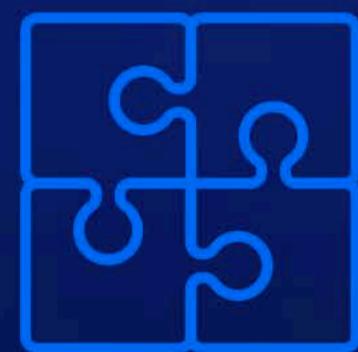


Educate, Integrate, Replicate,

Support Forever

proc_4() - Start Smart

Concepts across Company



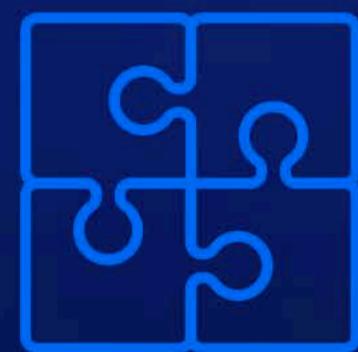
Educate, Integrate, Replicate,

Support Forever

- Deployment is Everything

proc_4() - Start Smart

Concepts across Company



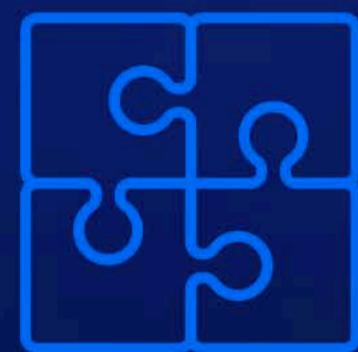
Educate, Integrate, Replicate,

Support Forever

- Deployment is Everything
- Start Small but End-to-End

proc_4() - Start Smart

Concepts across Company



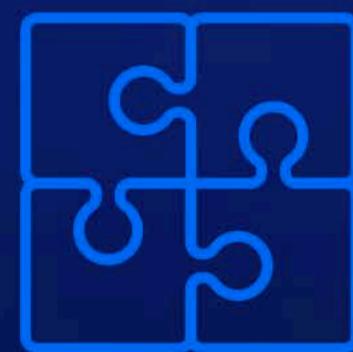
Educate, Integrate, Replicate,

Support Forever

- Deployment is Everything
- Start Small but End-to-End
- Analytics should match Experience

proc_4() - Start Smart

Concepts across Company



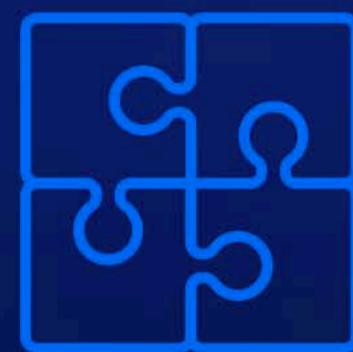
Educate, Integrate, Replicate,

Support Forever

- Deployment is Everything
- Start Small but End-to-End
- Analytics should match Experience
- Re-Gift Great Packages

proc_4() - Start Smart

Concepts across Company



Educate, Integrate, Replicate,

Support Forever

- Deployment is Everything
- Start Small but End-to-End
- Analytics should match Experience
- Re-Gift Great Packages

...DevOpsLite

end() - Here's an Example

```
function add(a, b) {
  return a + b;
}

function multiply(a, b) {
  return a * b;
}

function divide(a, b) {
  return a / b;
}

function subtract(a, b) {
  return a - b;
}

function calculate(operation, a, b) {
  if (operation === "add") {
    return add(a, b);
  } else if (operation === "multiply") {
    return multiply(a, b);
  } else if (operation === "divide") {
    return divide(a, b);
  } else if (operation === "subtract") {
    return subtract(a, b);
  } else {
    return "Unknown operation";
  }
}

// Example usage:
const result = calculate("add", 5, 3);
console.log(result); // Output: 8
```

```
function calculate(operation, a, b) {
  if (operation === "add") {
    return a + b;
  } else if (operation === "multiply") {
    return a * b;
  } else if (operation === "divide") {
    return a / b;
  } else if (operation === "subtract") {
    return a - b;
  } else {
    return "Unknown operation";
  }
}

// Example usage:
const result = calculate("add", 5, 3);
console.log(result); // Output: 8
```

end() - Here's an Example

First Customer

- 6 People - 4 weeks
- 0 projects to date
- Event Experience
- Multi-Platform
- Deployment

end() - Here's an Example

First Customer

- 6 People - 4 weeks
- 0 projects to date
- Event Experience
- Multi-Platform Deployment



end() - Here's an Example

First Customer

- 6 People - 4 weeks
- 0 projects to date
- Event Experience
- Multi-Platform Deployment



RedHat MAP to the Rescue

end() - Here's an Example

First Customer

- 6 People - 4 weeks
- 0 projects to date
- Event Experience
- Multi-Platform Deployment



RedHat MAP to the Rescue

- Provided Backend Services, Connectors and Deployment that powered v1.0

end() - Here's an Example

First Customer

- 6 People - 4 weeks
- 0 projects to date
- Event Experience
- Multi-Platform Deployment



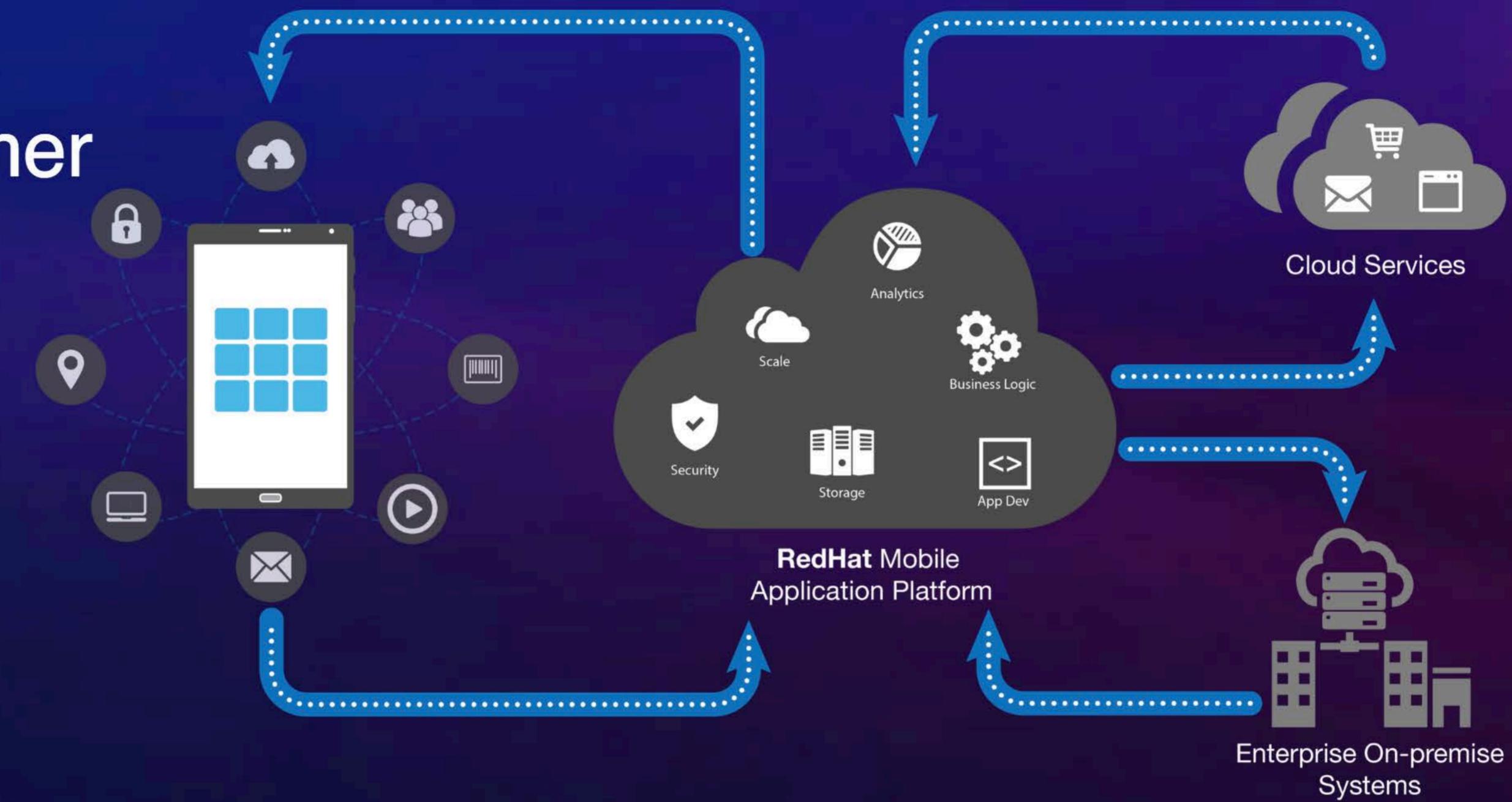
RedHat MAP to the Rescue

- Provided Backend Services, Connectors and Deployment that powered v1.0
- Security and Analytics for v1.1 and v2.0

end() - Here's an Example

First Customer

- 6 People - 4 weeks
- 0 projects to date
- Event Experience
- Multi-Platform Deployment



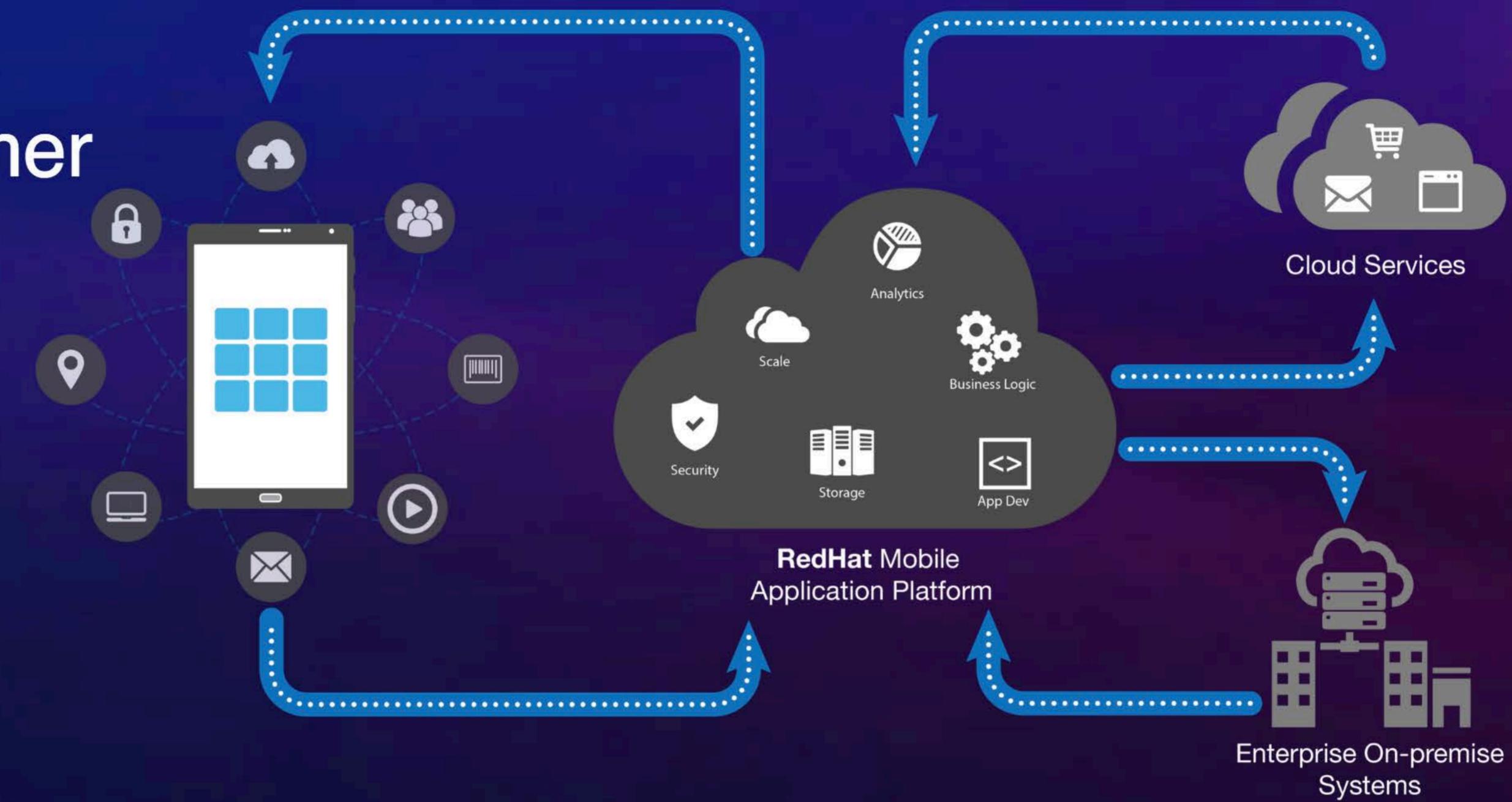
RedHat MAP to the Rescue

- Provided Backend Services, Connectors and Deployment that powered v1.0
- Security and Analytics for v1.1 and v2.0
- Flexibility to expand to v.7 (July 16)

end() - Here's an Example

First Customer

- 6 People - 4 weeks
- 0 projects to date
- Event Experience
- Multi-Platform Deployment



RedHat MAP to the Rescue

- Provided Backend Services, Connectors and Deployment that powered v1.0
- Security and Analytics for v1.1 and v2.0
- Flexibility to expand to v.7 (July 16)
- Profitable Product to build a DevOps practice around

end() - Here's an Example

First Customer

- 6 People - 4 weeks
- 0 projects to date
- Event Experience
- Multi-Platform Deployment



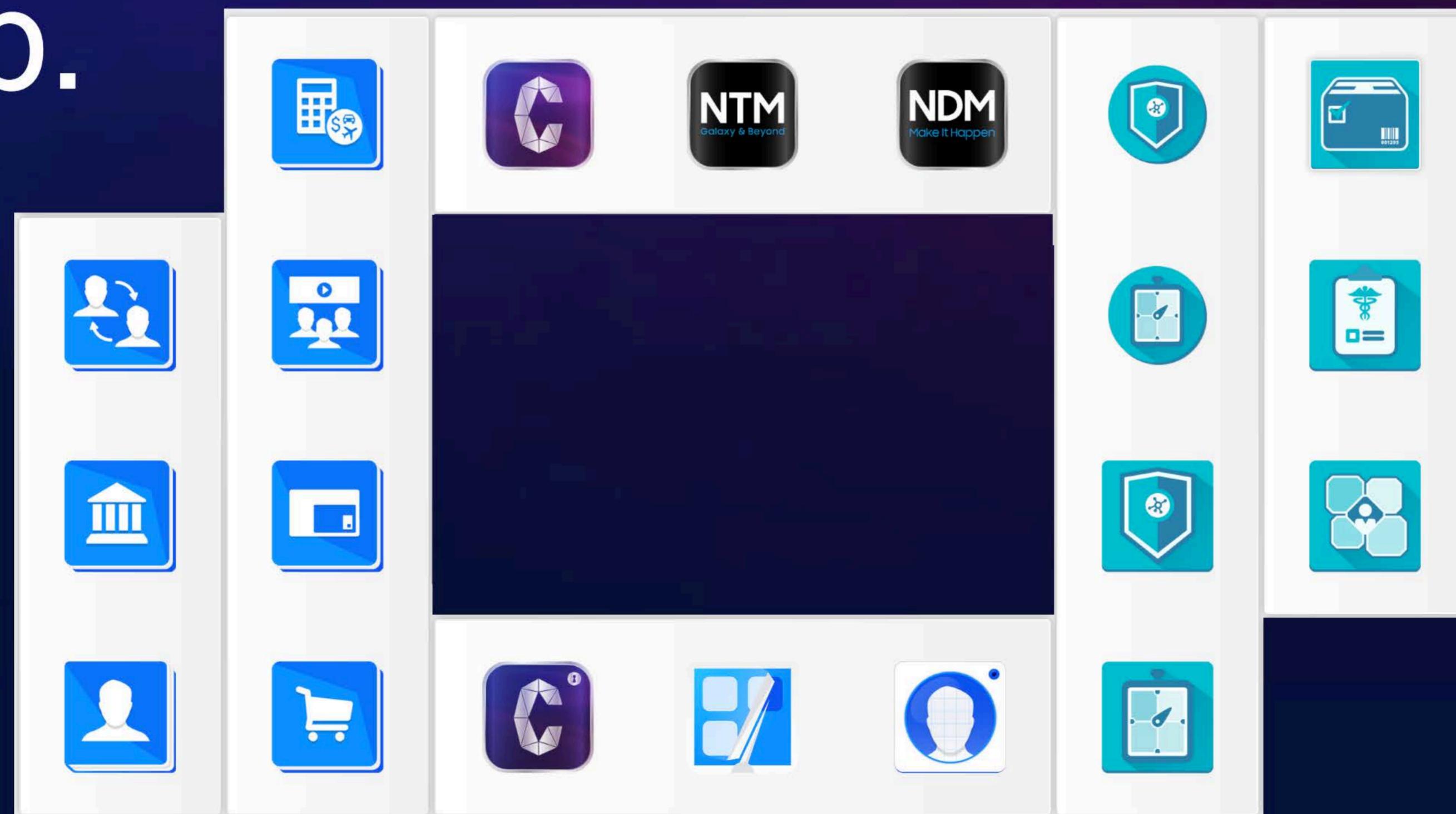
RedHat MAP to the Rescue

- Provided Backend Services, Connectors and Deployment that powered v1.0
- Security and Analytics for v1.1 and v2.0
- Flexibility to expand to v.7 (July 16)
- Profitable Product to build a DevOps practice around
- Leading to 3 Products, 15+ Releases, 35+ Demos in 18 months same 6 people

Question

Does DevOps Work?

Yep.



Thank You

Richard Balsewich

Samsung Business Services

Presentation template based on ImpressJS by Bartek Szopka
Some Icons made by Freepik from www.flaticon.com is licensed by CC 3.0 BY