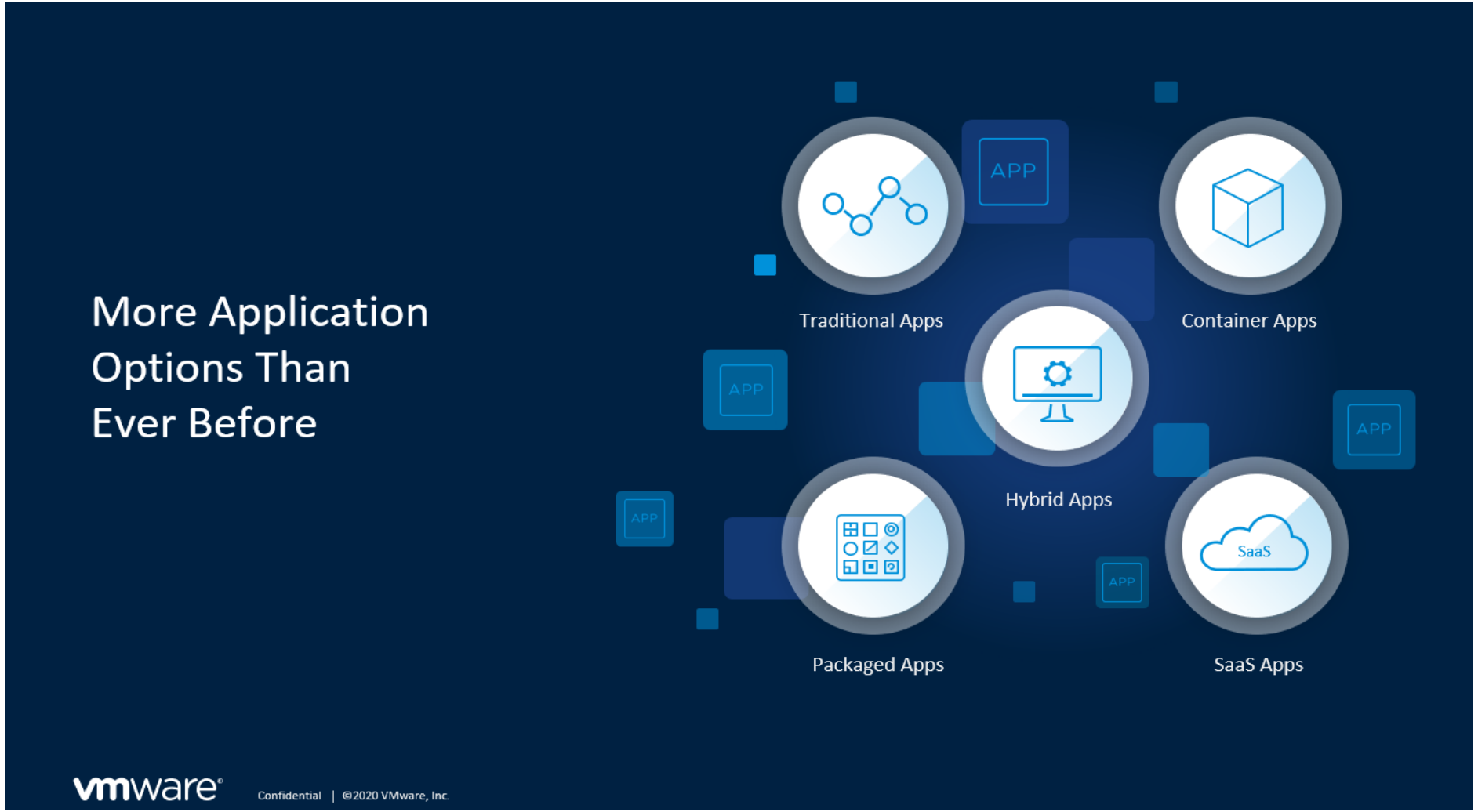
VMWare Launch - Working with Cloud

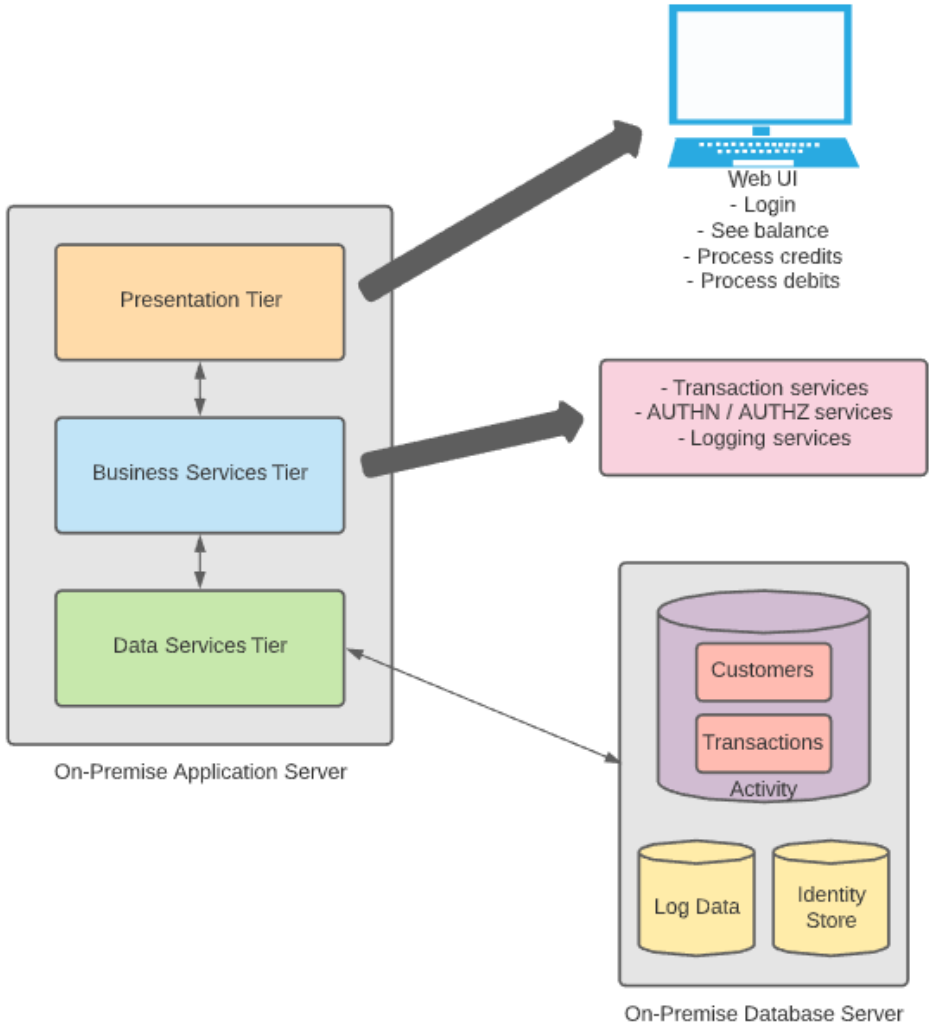
Application Options – Monolith vs. Microservices

Allen R. Sanders
Senior Technology Instructor

Class Roadmap



Banking App – Current State



Microservices

- One of the current trends in software development (though it's not really new)
- Built around concept of SOA (Service Oriented Architecture)
- Requires an approach that favors decomposition
- The architecture, design and testing concepts we've been discussing complement

Microservices – Key Characteristics

Develop Intelligence

- "Small" units of Service Oriented Architecture, or SOA
- Independently-deployable (KEY)
- By extension, independently-scalable
- Modeled around business domain (instead of tech domain)

Microservices – Key Characteristics

- Interact over networks (including the Internet)
- Technology can be flexible
- Encapsulate both business capability AND data
- Data sources are not directly shared but exposed through well-defined interfaces
- About cohesion of business functionality vs. technology

vs. Monolithic

Develop Intelligence

- Monolithic applications can exhibit some of the same characteristics as microservices:
  - Distributed
  - Interact via network
  - Might use unshared data sources
- Key difference, though, is requirement to deploy all parts of the system together (not independently-deployable)

Microservices – Benefits vs. Costs

Develop Intelligence

Benefits:

- Enables work in parallel
- Promotes organization according to business domain
- Advantages from isolation
- Flexible in terms of deployment and scale
- Flexible in terms of technology

Microservices – Benefits vs. Costs

Develop Intelligence

Costs:

- Requires a different way of thinking
- Complexity moves to the integration layer
- Organization needs to be able to support re-org according to business domain (instead of technology domain)
- With an increased reliance on the network, you may encounter latency and failures at the network layer
- Transactions must be handled differently (across service boundaries)

Microservices & Good Architecture

Develop Intelligence

- With microservices, key concepts:
  - Cohesion (goal to increase)
  - Coupling (goal to decrease)

- Our earlier discussions on SOLID and the twelve-factor app are quite applicable to targeting a microservices architecture:
  - Clean, logical organization of components
  - Maintainability
  - Clear boundaries, encapsulation and separation of concerns in the components used to build out complex systems
  - Techniques that minimize coupling
  - Being “surgical” with our change

Microservices & Containers

Develop Intelligence

- Microservices – with their smaller size, independently-deployable and independently-scalable profile, and encapsulated business domain boundary – are a great fit for containers
- Using Kubernetes (e.g. a VMWare Tanzu Kubernetes Grid), sophisticated systems of integrated microservices can be built, tested and deployed
- Leveraging the scheduling and scalability benefits of Kubernetes can help an organization target scaling across a complex workflow in very granular ways
- This helps with cost management as you can toggle individual parts of the system for optimized performance

Microservices & the Cloud

Develop Intelligence

- Microservices have broad support across multiple Cloud providers
- One option includes standing up VM's (IaaS) and installing / managing a Kubernetes cluster on those machines or
- Another option includes leveraging a managed service (PaaS) provided by the CSP
- Microservices are a great option for the Cloud because the elastic scalability provided by the Cloud infrastructure can directly support the independent scalability needed with a microservices architecture

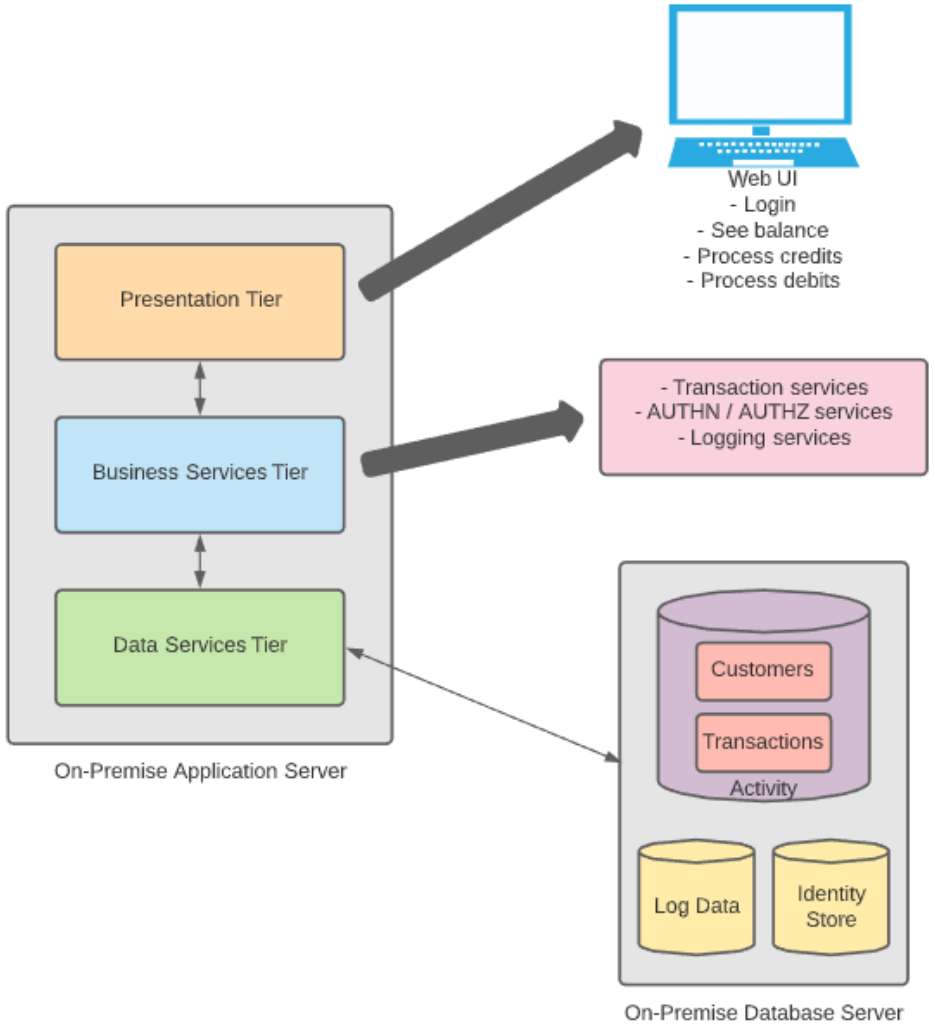
Microservices & the Cloud

Develop Intelligence

- In the banking example we discussed at the beginning of this session, without microservices (or at least the ability to break the monolith up into smaller components), Cloud migration will be “all or nothing”

- Microservices can increase feasibility of a migration to Cloud over time using a hybrid configuration

Banking App – Monolith Revisit



Brainstorming:

How might we break up this application into multiple business domains?

Do you see any good candidates for microservices here? If so, what are they and why?

Break (10 min.)

THANK YOU

Develop
Intelligence


Develop
Intelligence