Applications in the Cloud

Chapter 8

Cloud Applications Topics

Most cloud applications, such as Dropbox and Salesforce.com, are Software as a Service (SaaS).

Unit Topics

- Understanding the role of standard applications
- Developing cloud-ready applications
- Migrating applications to the cloud
- Preparing for technical challenges
- Identifying and mitigating risks

The Role of Standard Applications

- Here, the term 'standard application‘ describes any application that is not a cloud application.
- Standard applications can be viewed as having three logical tiers: presentation, application (or logic), and data.
- See next slide…

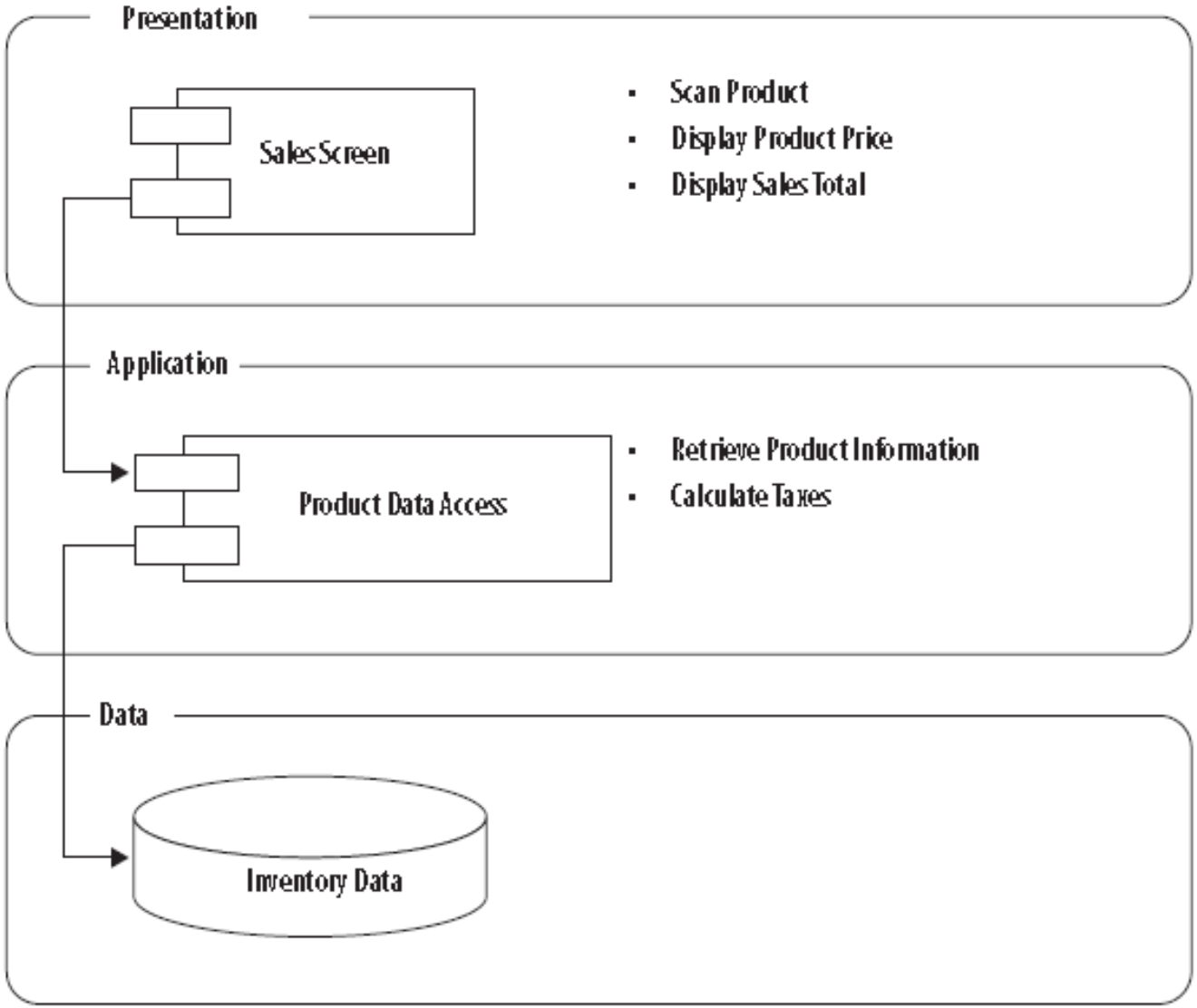
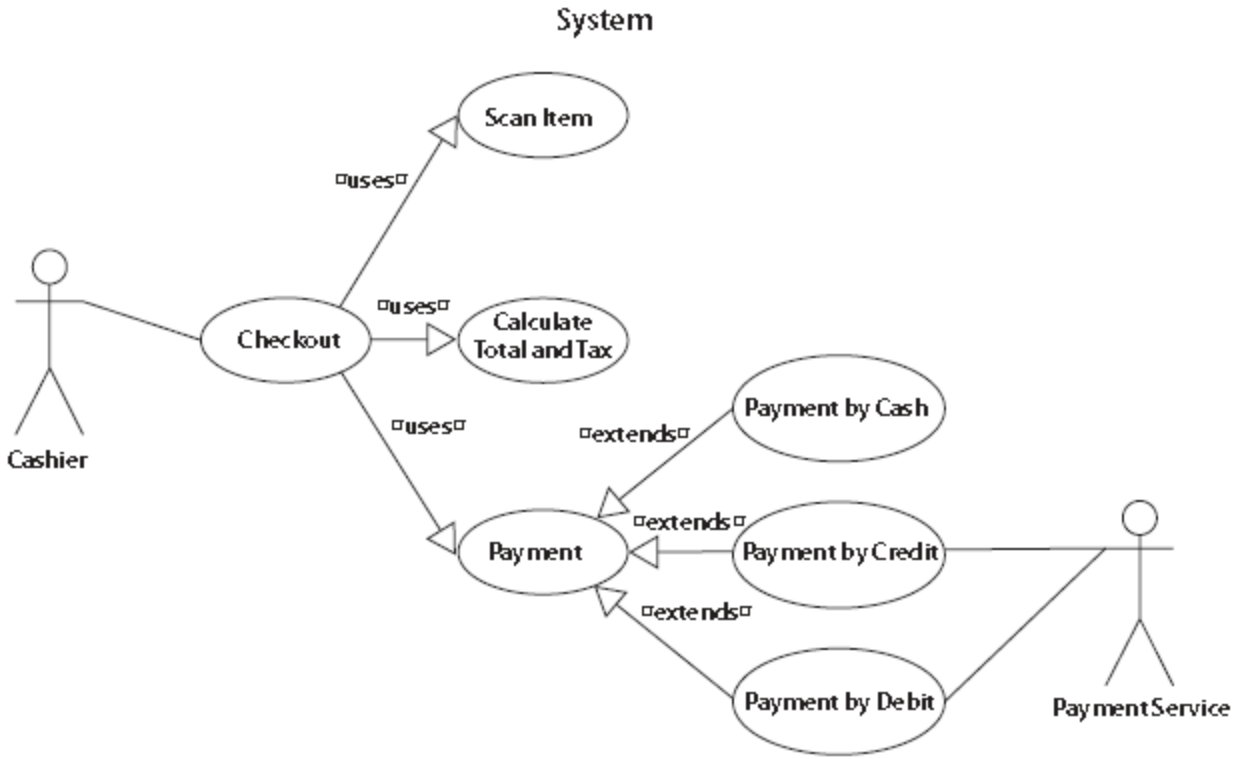


FIGURE 8.1 An example of the logical tiers in a point-of-sales application

Sample Detailed Use Case for a Checkout Service



Use Case Is Broken Down Into Three Simple Operations:

1. Scan item
2. Calculate total and tax
3. Payment.

The very first operation, scan item, can be described as:

1. Clerk is presented with the option to type the item code or scan it.
2. System retrieves description, unit price, and tax information for the item.
3. System updates current list of items in sale and calculates current total.
4. System displays current item information and current total.

Taken this single case study, you can construct an application using logical layers.

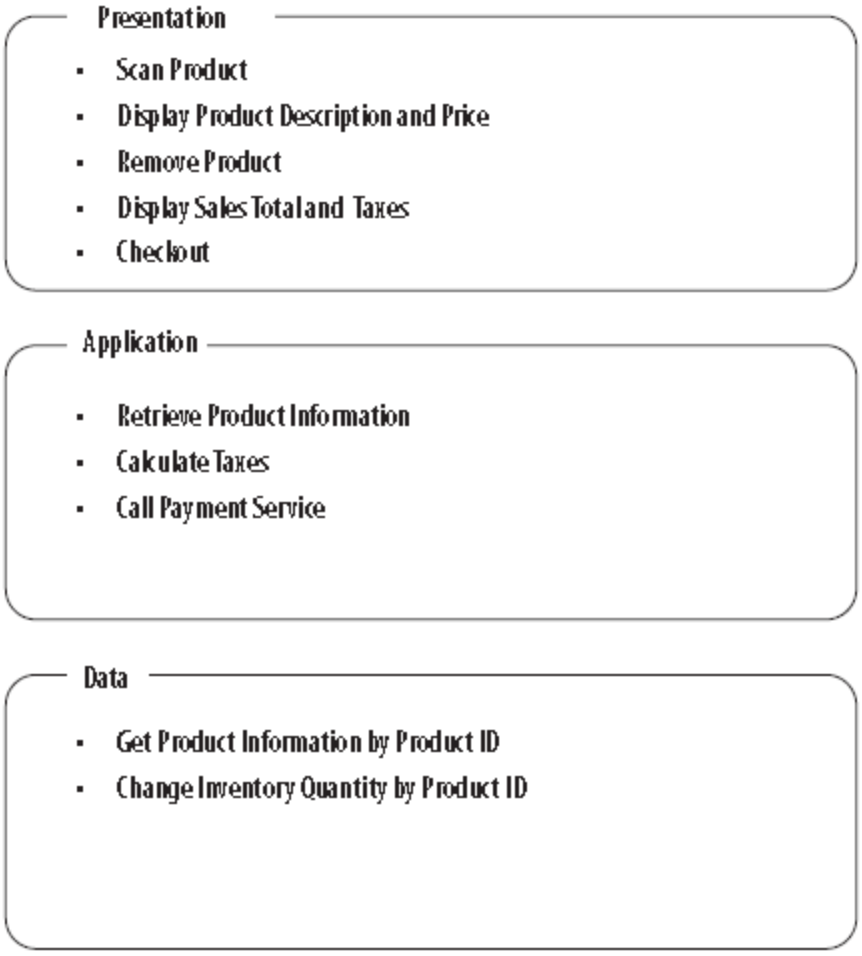


FIGURE 8.3 Checkout use case functions broken into layers

Point of Sale Application

- There is a lack of context on this simple point-of-sale application.
- Is it used by a single user in a small shop, a local supermarket, a popular chain, or even an online store? Without the context, there is no way of deciding the actual physical structure of the application.
- Each one of these different contexts will drive the application design to a different physical structure, such as desktop, distributed, web based, or cloud.

Desktop Applications

- Desktop application can use all application programming interfaces (APIs) made available by an operating system to ensure that its look and feel are familiar to users.
- The fact that the data consumed is not shared with other users or applications allows developers to create a desktop application faster than, for instance, the type of application that might require concurrent access to data, sharing of information, and communication across a network.
- Does not require access to the Internet or any other network, interaction with another user, or access to any other network service.

Desktop Applications

- Will not end because of cloud computing.
- Will still be needed in situations where access to external data or services is not needed or possible.
- Some applications are better off left in a desktop environment.
- In the case of the hypothetical POS application described previously, if the application is to be used in a small shop, with a single point of sale, all three logical layers could run on the same point-of-sale computer.
- A single application could provide the user interface with regular elements of a common operating system and the logic within the same process.
- This single application could also contain code that accesses data stored in the computer hard drive in any format the application is able to read and write to.

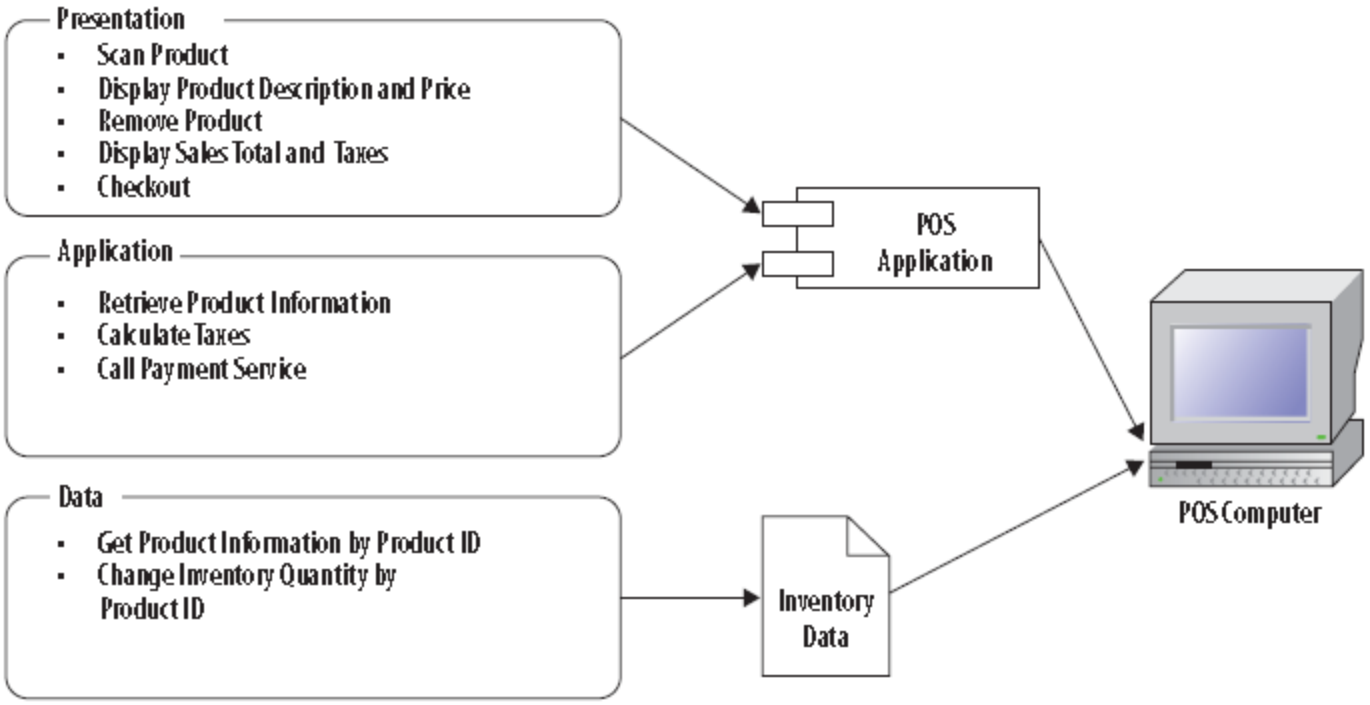


FIGURE 8.4 Representation of the POS application as a desktop application

- Data layer would contain a list of all products sold in the store along with their unit price and available quantity in stock.
- When a product is sold, the application retrieves its information and, once the sale is complete, decreases the inventory.

Distributed Applications

- True cloud applications must be designed as distributed applications.

For now, take our small shop with a single cashier, and imagine sales are going well.

- Shop is growing.
- Lines forming at the single point of sale.
- Something must be done to decrease line or customers will shop elsewhere.
- New point of sale is needed.
- But what if sales keep growing and more customers come in?
- Maybe two points of sale are not enough. More might be needed.
- Application design must change to allow use of multiple points of sale
  - And possible future growth.
- Currently data is stored in a single file, in the one and only computer available.
  - All the inventory data is listed there.
- Any point of sale that's added will need access to this data.

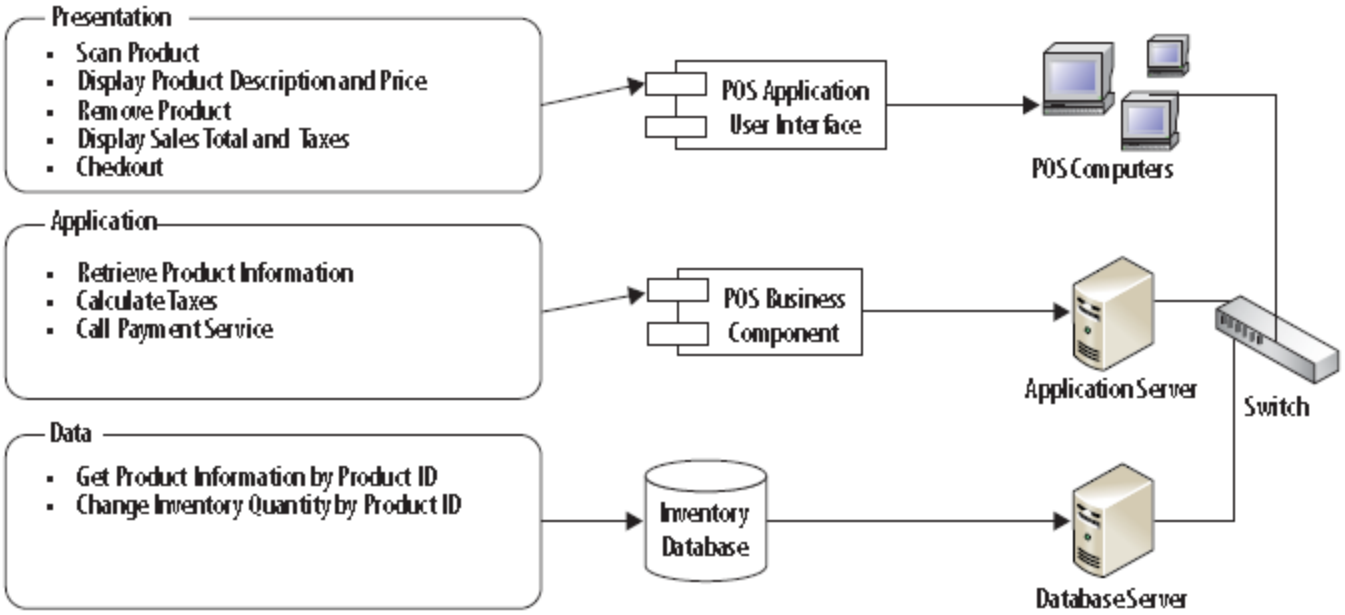


FIGURE 8.5 Representation of the POS application as a distributed application

- Data needs to physically separate from the rest of the logical tiers.
- Because the new points of sale will still be in the store, store can decide what type of operating system to use and maintain the one currently in use, preserving the user interface.
- Data must be now placed on a separate physical computer, accessible to all points of sale.
- Logic can run on the computer storing the data, on the computers used as points of sale, or on a separate computer dedicated to the application tier, depending on processing and scalability needs.

Distributed Application Design

- Requires availability and scalability design considerations.
- Single points of failure should be avoided.
- Several techniques for availability.
  - Each point of sale could have a cached copy of the data.
    - Then, if the server hosting the data is not available, a point of sale can still be used to make a sale as long as the data it contains cached is trusted.
  - All sales would be stored locally until the data server is back online.
  - At this point, data from all points of sale can be copied to the data server and synchronized.
- Another solution would be to use failover clustering.

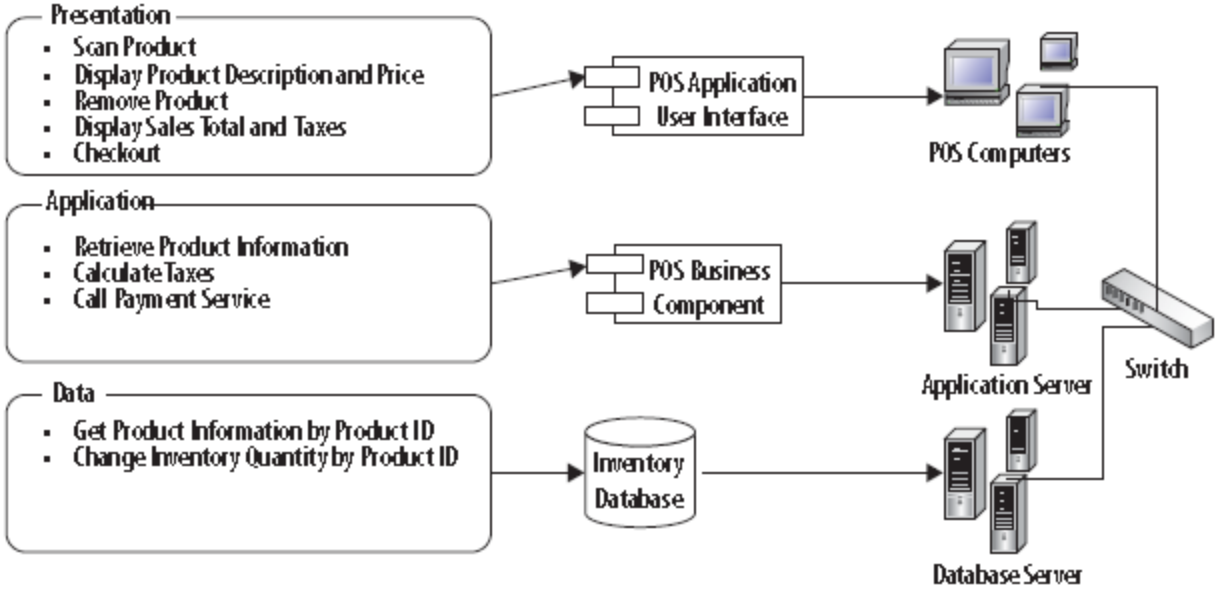


FIGURE 8.6 Representation of the POS application as a highly available distributed application

- Failover clustering would not require changing any of the code for the application but has a higher hardware cost.
- For an example of a highly available multitiered application.

Web-Based Applications

- Standard distributed applications can solve most high availability and scalability needs.
- Most applications today are standard distributed applications.
  - There will always be a place for standard distributed applications.
  - However, standard distributed applications require control of the entire enterprise architecture.
- If devices with different operating systems are needed in the presentation layer, a different user interface for each must be designed and developed.
- The APIs used to communicate between the physical layers may also require configuration of firewalls.

POS Example

- Imagine that sales are booming.
- All of a sudden customers are calling in from other states asking about products, wanting to purchase them.
- Existing inventory is not enough to handle these new orders.
- Deals are struck with suppliers to allow accessing their existing inventory when selling products that are not available in the store.
- And to help serve out-of-town customers, a web storefront must be created.
- Once more, the entire physical structure must change.
- Point of sale needs to be available over the Internet for the customer.
- There is no way of limiting what type of computer or operating system that customer is using.
- And there is no way to train these new customers on the existing point-of-sale application.
- An interface that's common to all must be used, something that is available to anyone who has access to the Internet.
- The presentation layer must change.

Cloud Applications

- Years go by, and the once-small shop with a single point of sales now generates thousands of transactions a day.
- Customers from all over the globe flock to the company's website to order.
  - Web-based application holds up well but requires maintaining hundreds of servers.
  - IT costs reflect that maintenance.
- Company decides to invest in a multimillion-dollar TV marketing campaign during the upcoming Super Bowl.
- Hard to estimate the number of users who will access the website during and after the massive TV campaign.
  - But the site must stay online and accommodate for the increasing number of users.

- Ultimate cloud application scenario. Goals:
  - Reduce the number of on premises servers
  - Automated scalability.
- Once again, the physical design of the once-simple POS application must change.

Developing Cloud-Ready Applications

- Not every application should be migrated to the cloud.
- Important to identify which types of application will benefit from cloud computing and then ensure that those applications are designed to be cloud-ready.

Cloud-Ready Application Patterns

- The main technical characteristic of a cloud-ready application is the need for elasticity.
- You should automatically scale out when usage is high to accommodate the required compute needs and scale down when compute needs decrease, cutting costs.

4 Main Cloud Ready Application Patterns

1. Start Small, Grow Fast
2. Predictable Burst
3. Unpredictable Burst
4. Periodic Processing

Start Small, Grow Fast

- Typical scenario for startup companies.
- Scalability is vital if the product goes viral, yet investment should be minimized.
- Figure shows the relationship of resource consumption and time in a start small, grow fast design pattern.

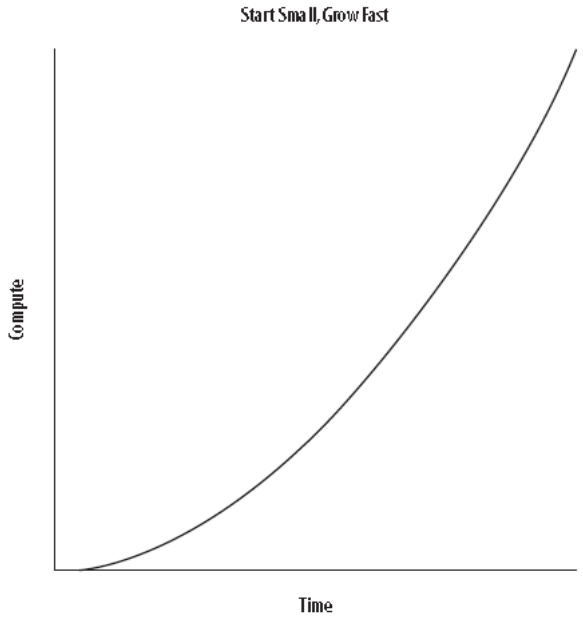


FIGURE 8.9 Start small, grow fast

Predictable Burst

- A new release of a product (think iPhone) or marketing campaign (Super Bowl) can cause an e-commerce application to suffer a burst that is both welcome and predictable.
- Often linked to single event.
  - Very possible that cloud computing may be used for just this one-time event.
- Figure 8.10 shows the relationship of resource consumption and time in a predictable burst design pattern.

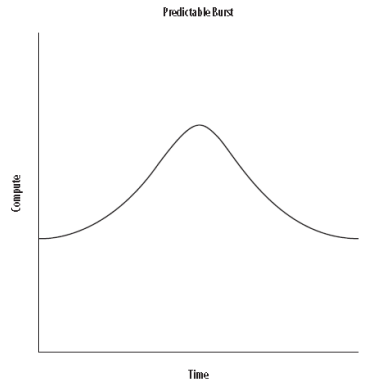


FIGURE 8.10 Predictable burst

Unpredictable Burst

- Very similar to the predictable burst, without association of the burst to an event.
- Figure 8.11 shows the relationship of resource consumption and time in an unpredictable burst design pattern.

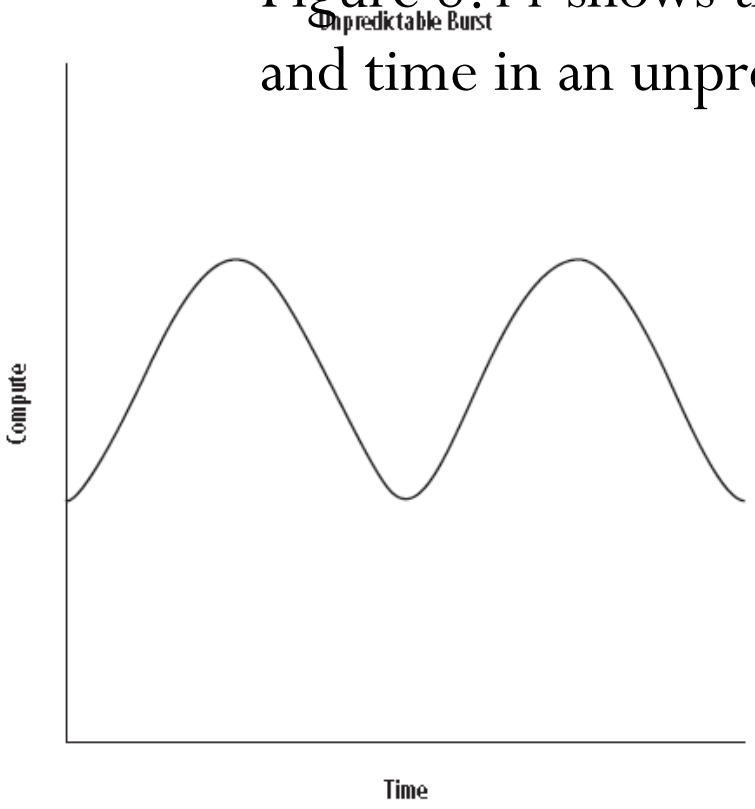


FIGURE 8.11 Unpredictable burst

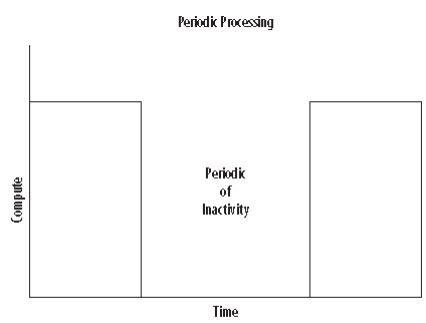


FIGURE 8.12 Periodic Processing

Periodic Processing

- Many enterprises have applications that are heavily used only during a specific time period.
- Applications include tax processing and election voting.
  - Payroll processing and annual review.
- Amount of compute time required for such applications in such a short period of time does not justify the investment on an infrastructure that will be left without use for long periods of time.
- Figure 8.12 shows the relationship of resource consumption and time in a periodic processing design pattern.

Cloud-Ready Application Development

- Cloud-ready applications can exist on a standard distributed environment but also take advantage of the benefits of cloud computing.
- Two main factors for designing a cloud-ready application:
- Stateful vs. stateless applications
  - Stateful applications require information about objects to be maintained between server calls.
  - In a distributed environment, especially in a cloud environment, there is no guarantee that the same server will answer subsequent requests from a client, consequently stateful objects should be avoided at all costs.
- IaaS vs. PaaS
  - No PaaS-based application standards.
    - Each provider uses different API.
  - Choosing a specific provider might force a lock-in with a technology that cannot be migrated later to a different provider.
  - Use IaaS unless you are comfortable with the technology used by a PaaS provider and you do not foresee a change in the technology used.

Pricing plan

Providers offer:

- Pay-as-you-go plans
- Monthly plans
- Yearly plans
- Or any combination.
- Thoran Rodrigues published an article on TechReplubic early in 2012 showing that the monthly price for a 1 CPU, 2 GB RAM cloud server at that time varied anywhere from US$40 to US$270 for 720 hours (24 hours for 30 days in a month).
- Full study can be found at the following location:

http://www.techrepublic.com/blog/the-enterprise-cloud/11-cloud-iaas-providers-compared/5285/

Service-level agreement (SLA)

- Providers guarantee an SLA of anywhere from 99.9 percent to 100 percent.
  - Be aware of 100 percent SLAs and ensure that financial guarantees are in place if the provider is not able to deliver the SLA specified by contract.

Number of data centers

- Smaller providers like ReliaCloud, GoGrid, and Bit Refinery have one or two data centers
- Providers like Rackspace, Amazon, Terremark, and Go Daddy have more than five data centers spread all over the world.

Certifications

- Ensure that the provider has any certifications required for your application, such as Payment Card Industry Data Security Standard (PCI DSS) or Statement on Standards for Attestation Engagements No. 16 (SSAE 16) and Statement on Auditing Standards No. 70 (SAS 70).

Support

- Some providers have extensive support over the phone, while others only handle support tickets online and maybe be slow to respond.

Monitoring

- Level of monitoring varies.
- Some providers do not have any built-in monitoring, requiring the installation of third-party tools and extra services, while others have integrated monitoring tools available at no extra cost.

Instance types

- Most providers have a set number of servers that can be used, with a specific number of CPUs, amount of memory, and operating system.

Others have fully customizable instances.

- Data transfer cost Most providers charge for outbound data transfer; some also charge for inbound data transfer.

Migrating Applications to the Cloud

- Most candidates for cloud-based applications are already being used in the enterprise.
- Looking back at the cloud-based application patterns discussed earlier in this chapter and reflecting on the existing applications in the enterprise architecture of any organization will result in a list of several applications that are good candidates for cloud computing.

Cloud Migration

- Once applications are identified, then needed to research how they can be migrated to the cloud.
  - Some applications can be completely replaced by existing SaaS applications
  - Others can be easily migrated to an IaaS provider, and a few can take advantage of existing PaaS offers.
- It is important to identify the type of service to use because it will affect how the application might need to change and associated costs for maintaining it once it's migrated.
- Table 8.1 (next slide) describes the migration choices and their differences.

TABLE 8.1 Migration choices

Migration to	Pros	Cons
SaaS	Least cost Replaces current application with existing SaaS offering	Less flexibility for customization
PaaS	Lower cost than IaaS using comparable operating system and support No operating system maintenance	Provider technology lock-in Changes to existing application
IaaS	Minimal code change to application Use of familiar development technology	Operating system maintenance

- Different IaaS and PaaS vendors provide guidance on migrating existing applications to their environment. Here are some of these guides:

Windows Azure:

- www.microsoft.com/en-us/download/details.aspx?id=29252

Amazon EC2:

- aws.amazon.com/documentation/ec2/

Rackspace:

- rackspace.com/knowledge_center/article/rackspace-open-cloud-migration- considerations

Technical Challenges

Not every application is fit to be a cloud application.

- Distributed applications that require automated scalability and high availability and are more CPU bound than I/O are applications that can easily benefit from cloud computing.
- The basic rule of thumb
  - Applications that process large amounts of data and are I/O bound should remain on premises;
  - Applications that require processing small amounts of data and are CPU bound can benefit from cloud computing.
- The reason for the CPU-bound versus I/O-bound decision lies in one of the main technical challenges found with cloud-based applications: moving data over the Internet.
- Bandwidth can become very expensive when large amounts of data need to be transferred in and out of a cloud-based application.
  - For example, Amazon charges US$0.12 per GB of data transferred out of its data centers.
- An application that generates 10 TB of data a day will cost US$1,200.00 a day just on data transfer.
- Not only is that a financial challenge, it is also a technical challenge to transfer that amount of data over the Internet in places where connectivity to the Internet may still rely on analog dial-up lines.

Cloud Based Application Challenges

Big data

- Big data applications generate several terabytes of data a day.
  - For example, eBay generates over 150 TB of logging data every day.
- Moving this out of a public cloud can cost hundreds of thousands of dollars a month.

Unstructured data

- Flat files tend to require a lot of processing for parsing data into a more manageable format.
  - Whenever possible, structured tabular data should be used.

Security

- Personally identifiable datasets and trade secrets require protection.

Compliance

- Certain countries do not allow for personally identifiable data (PID) to cross geographical boundaries.
- A common issue in the European Union. Cloud providers duplicate data across their data centers, causing compliance issues for PID.

Learning curve

- Software architects and developers need to be trained in the development of cloud-based applications and may be required to learn proprietary APIs to create new applications in a PaaS environment.

Identifying and Mitigating Risks

- Risk is defined in Information Technology Infrastructure Library (ITIL) 2011 as the possibility that an event will occur and affect the ability to achieve an objective.
- Step in the risk management process:

Risk identification

- Anyone in the enterprise should be able to present a basic risk statement in the form of a simple sentence stating that IF a certain event occurs THEN a specific objective will not be met.

Risk classification

- Risk management team responsible for looking at identified risks and classifying them by analyzing their root cause, possible outcome, and type of risk (availability, integrity, performance, security, etc.).

Risk prioritization

- Different risk management processes prioritize risks differently.
  - Simplest way to prioritize risks is to attribute a value to the probability of the risk happening and the impact of the risk to the organization.
  - Assigning numeric values to these factors will allow the multiplication of one by the other to come up with a risk factor. The higher the risk factor, the more dangerous the risk.

Risk planning

- Once risks have been prioritized, it is necessary to respond to them.
- Some risks can be avoided by changing how the associated task is performed.
- Some can be mitigated by changing the scope of the task
- Some might simply be accepted and a contingency plan created in case they occur.

Risk monitoring

- Top risks for the enterprise should be monitored closely
  - Triggers must be defined to allow the enterprise to identify when they happen and roll out its contingency plan if necessary.

Some cloud computing associated risks

- Vendor lock-in
  - Many cloud service providers offer development tools that are proprietary and work exclusively within their cloud environment.
- The more applications an organization develops with these tools, the more the organization is locked in with the provider.
  - Making it harder to move to a different provider.

Security and compliance

Some organizations are required to comply with regulations and laws such as:

- Sarbanes-Oxley Act of 2002
- USA Patriot Act
- Health Insurance Portability and Accountability Act of 1996 (HIPAA)
- EU Data Protection Directive.
- Depending on the services offered by the cloud service provider, the organization might not be able to obtain the necessary security incidents logs required by these regulations.

IT organizational changes

- If cloud computing is highly adopted and in consequence the IT personnel is drastically reduced, morale among the remaining members of the IT staff could be at risk.

Cloud Service Provider Maturity

- Most cloud service providers are young companies or represent a new line of business for well-established companies.

Reliability and performance issues

- Cloud service providers offer SLAs.

High-profile targets

- Well-known cloud service providers are high-profile targets for cyber attacks. The more high-profile customers they have, the more likely it is for hackers to try to break into their systems.
- Over-scalability due to DDOS Distributed denial of service (DDOS) attacks are hard to identify and can be seen as legitimate attempts to access an application.
- If there is no limit in the number of instances an application can bring online to allow for scaling out, a well-designed DDOS attack might inflict a costly penalty on the scalability of an application by spawning several virtual machines and increasing compute time.

Questions???