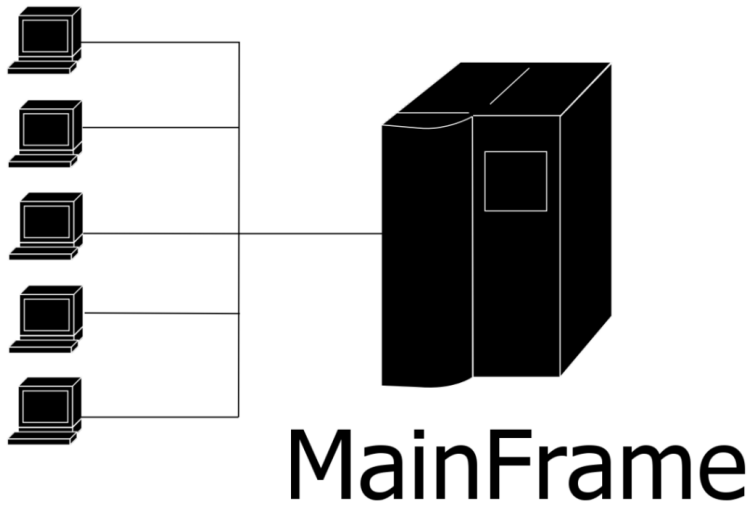
The need for XML Web Services

Nicky Randles – B00058026

Evolution of Distributed Systems

- Before we had PCs, distributed applications didn't really exist.
- Back then, using a PC involved a terminal interacting with a mainframe
- Terminals could be located in many different buildings
- The central computer did all the processing



Evolution of Distributed Systems

- As data storage and processing became more popular new application design patterns known as distributed applications became apparent.
- Unlike traditional applications that rely on a single system to run, distributed applications run on multiple computers within an a network.
- Two concepts that emerged with distributed applications:
  1. Distributed applications as service providers
  2. Distributed applications relationship with the web

Problems with traditional distributed applications

- Developing new distributed applications required new design techniques and models.
- They had all sorts of new problems such as:
  1. Client or Server failure – Handling loss of communication
  2. Different data types – May not be compatabile
  3. Security – Secure communication between client and server

Common XML Web Service Scenarios

- Common XML web service scenarios may include:
  1. Application service providers - XML web services are a good solution for designing applications that are meant for hosting. ASPs host applications that they then rent to subscribers
  2. Accounting and Finance – SOAP based web services are commonly used in business scenarios such as banking.

- Examples of web service providers are: Amazon and Google

Sample Question

May 2013, Question 3 Part B

While solving many problems, the advent of distributed applications leads to new challenges. Discuss.

Answer:

There are many challenges that can arise while dealing with distributed applications. Common challenges involve communication failure, data types, and security. Most of these problems have arose because we are now communicating remotely rather than locally. The client and server may lose connection with one another which may lead to the loss of data. It is important that the server knows how to handle these connection failures. Distributed applications may run into problems with their data types as the may not be compatible across different operating systems. Therefore, it must be considered on how to handle certain data types which are not compatible on some operating systems. In distributed applications there are more opportunities for security threats. Authentication and authorization methods and ways to secure the communication between the client and server must be considered. Also methods to prevent attacks such as man-in-the-middle attacks, denial-of-service attacks and replay attacks must be considered.

Questions?