ECE 315 Assignment 1

Arun Woosaree
XXXXXXX

February 1, 2020

1

Hard real-time vs. Soft real-time embedded systems

Similarities

- With both, we want to avoid violating real-time constraints

Differences

- Violations of real-time constraints are undesireable, but tolerated in soft real-time systems while in hard real-time systems, violations of real-time constraints are unacceptable (a violation for a hard real-time system would cause catastrophic failure, or even death.)

When selecting an implementation strategy, it is useful to distinguish between hard and soft real-time systems because you want to select a strategy that is appropriate. You want to consider what might happen in the case of failure to meet real-time constraints. If a violation of these constraints would cause catastrophic damage, such as loss of life, you have a hard real-time system, so your implementation strategy should take that into account. If you have a soft real time system, and failure to meet the real-time constraints is not catastrophic, your implementation strategy does not need to avoid failure to the same extent as it would need to for a hard real-time system.

For example, you probably would have a different implementation strategy for making a pacemaker versus a vending machine. In one scenario, a person's life is dependent, while in the other, a person may or may not get a snack.

2

2.1 Cloud Computing

As per ISO/IEC 17788:2014 :

" Cloud computing is a paradigm for enabling network access to a scalable and elastic pool of shareable physical or virtual resources with self-service provisioning and administration on-demand. The cloud computing paradigm is composed of key characteristics, cloud computing roles and activities, cloud capabilities types and cloud service categories, cloud deployment models and cloud computing cross cutting aspects that are briefly described in [clause 6 of ISO/IEC 17788:2014]. "

Advantages:

- The cloud computing provider takes care of purchasing and maintaining hardware, not the user

- Allows for flexibility. The user can scale services as they need on-demand. That way, a user only pays for the resources they use. For example, the user can request more computing resources when their service has more traffic, but when there is less traffic, the user can scale down their operations to save cost

- Reduces barrier of entry for deploying an application. One can deploy a toy project they've been working on for a few dollars, without investing in the up-front cost of purchasing a mainframe, maintaining it, and selling the computer when they're done.

Overall, I think this term is extremely well-defined, considering there is an ISO specification that is 10 pages long, and is very thorough in defining everything. It is, after all the international body for defining standards...

2.2 Edge Computing

According to Cloudflare:

" Edge computing is a networking philosophy focused on bringing computing as close to the source of data as possible in order to reduce latency and bandwidth use. In simpler terms, edge computing means running fewer processes in the cloud and moving those processes to local places, such as on a user's computer, an IoT device, or an edge server. Bringing computation to the network's edge minimizes the amount of long-distance communication that has to happen between a client and server. "

Advantages:

- Lower latency (better user experience, streaming and gaming applications for example. Users would experience higher quality streams and faster response times for games)

- Less bandwidth and server resource usage, and therefore lower costs associated with these resources

Overall, it is not very specifically defined. However, most sources seem to agree that the general idea is moving cloud computing resources geographically closer to users.

2.3   Fog Computing

According to the paper "Finding your way in the fog: Towards a comprehensive definition of fog computing" by L. M. Vaquero and L. Rodero-Merino:

> "Fog computing is a scenario where a huge number of heterogeneous (wireless and sometimes autonomous) ubiquitous and decentralised devices communicate and potentially cooperate among them and with the network to perform storage and processing tasks without the intervention of thirdparties. These tasks can be for supporting basic network functions or new services and applications that run in a sandboxed environment. Users leasing part of their devices to host these services get incentives for doing so."

Advantages:

- even lower latency and utilizing resources like bandwidth more efficiently compared to edge computing. It brings the edge closer to the user, kind of like how fog is closer to the ground than clouds

- allows for data to be processed locally for faster proccessing and response times. Applications include a "hive mind" of self driving vehicles, or sensors in a factory which need to automatically shut down equipment if something goes wrong

2.4   Mist Computing

Definition:
   Advantages:

- .

3

4

5

6

7

8

9