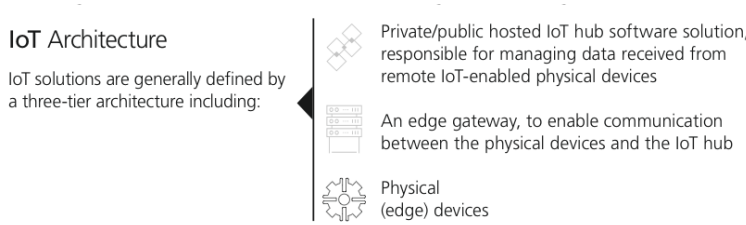
Solution Architecture Strategies for IoT in Medical Devices

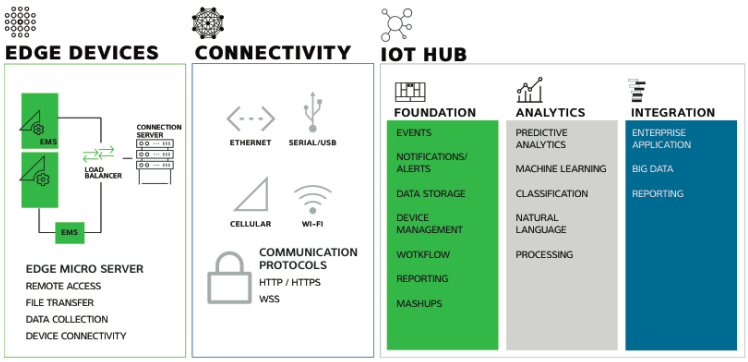
by Wayne Posner and Jordan Reynolds



Implementing an IoT solution can be a challenging initiative.

Implementing an IoT solution within an industry bound by numerous rules, regulations, and restrictions can add significant levels of complexity. Fortunately, with a well-planned and carefully architected solution, an IoT initiative can be designed to work within the rules and regulations. This means that businesses, such as those within the medical device industry, can begin to realize the enormous benefits gained through this transformative strategy.

Depending on the scale of the IoT implementation, there may be other architectural components such as load balancers and connection servers.



Medical Device Architectural Considerations

Within the medical device industry, there are additional architectural considerations.

- Does the device store patient data or can patient data be derived from examining device log data? If so, HIPPA compliance is a factor and must be addressed.
- Are there any legacy devices lacking networking capabilities?
- Does the environment in which the device is utilized implement any sort of network? If so, are there network security restrictions that preclude the device from connecting to the network?

- Are there government regulatory export restrictions that limit how where device data can be accessed? If so, a federated architecture with user location based security must be considered.

Security and encryption is always a key architectural consideration for any IoT solution. Many companies have strict constraints restricting specific data to users within specific organizations. Files transferred between the IoT hub and remote devices should be protected by strong encryption algorithms compliant with FIPS 104-2, while in flight, and validated against at least a CRC-256 checksum (SHA-256 hash is preferred), while at rest, to ensure malware has not been introduced.

Most medical devices deployed within a hospital environment are currently restricted from operating in an “always on” connectivity state. These devices operate in an “offline” mode. Working with these devices is a manual process. The typical use case for communicating with offline medical devices is that a qualified and/or authorized user will physically establish a connection between the device and a laptop or tablet. The laptop or tablet will allow the user to interact with the device data using custom software. Finally, the laptop or tablet will sync the device data with a custom server solution once it establishes network connectivity.

IoT Solutions for Medical Devices Operating Offline

How can an IoT solution help to optimize this type of manual process? There are three possible solutions that work within the confines of devices operating in an offline mode.

Embedded Edge Micro Server

Install an edge micro-server directly on the device, provided the device has built-in networking capabilities; when a connection is required between the device and the IoT hub, temporarily enabling network connectivity is the only requirement. This solution is the most ideal and requires the least amount of customization; however, network security protocols must allow for medical devices to temporarily establish network connectivity when necessary.

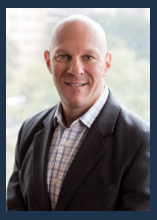
Attached Edge Micro Server

Install an edge micro-server on a computer that is always connected or can quickly be connected to the device via a serial or Ethernet connection. Some medical devices may lack network connectivity or the additional resources required to run an edge micro-server, or may not run an operating system compatible with micro-server frameworks such as Java, C, or .Net. A small computer running Linux can be used to host the edge micro-server and establish a serial connection with the medical device. Like the first solution, network security protocols must allow for this computer to temporarily establish network connectivity so that it can communicate with the IoT hub.

Custom Software

Create/Update custom software running on the laptop and/or tablet to enable it as an edge device that communicates with the IoT hub while also adding enhanced cache management functions. This allows the IoT hub to automatically receive cached data from the laptops and tablets (with an active network connection) and push updates down to be cached until the laptop and/or tablet reconnects with the device. Of the three solutions, this is the most complex and time intensive due to the amount of required programming.

Implementing a robust and scalable IoT solution requires a well-planned and thoughtfully designed architecture. Highly regulated



Wayne Posner

wayne.posner@kalypso.com

Wayne is a Senior Consultant in Kalypso's Digital Practice based out of Las Vegas, NV. When not architecting IoT solutions for Kalypso's clients, he can be found doing something adventurous or behind the lens of his camera photographing music icons.

Jordan Reynolds

jordan.reynolds@kalypso.com