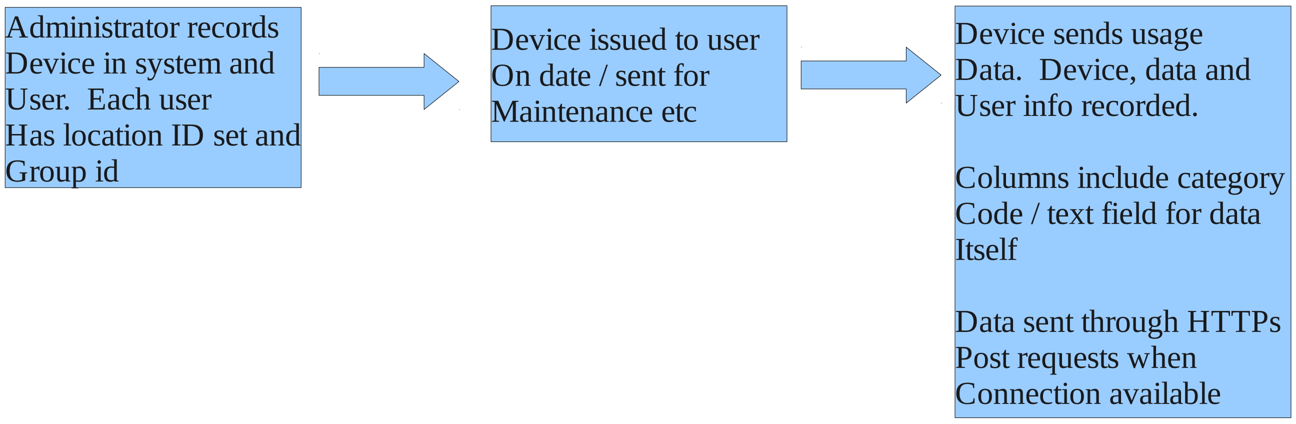
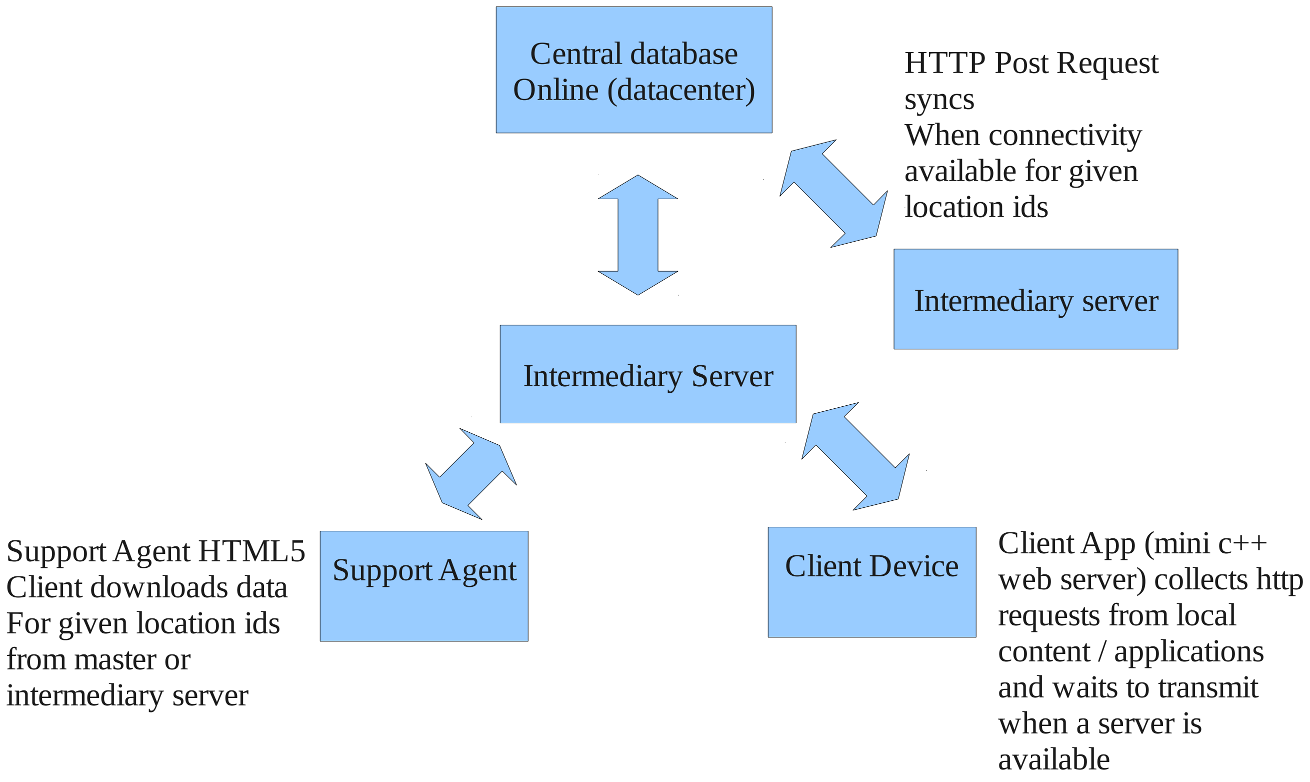
Distributed Inventory Database System

Overview

PAIWASTOON seeks a developer to create a distributed database for inventory of electronic equipment issued to users (laptops, tablets, etc).


Administrator records Device in system and User. Each user Has location ID set and Group id
Device issued to user On date / sent for Maintenance etc
Device sends usage Data. Device, data and User info recorded.
Columns include category Code / text field for data Itself
Data sent through HTTPs Post requests when Connection available

The system is distributed across areas of poor connectivity and therefor has to comprise of a central server, an intermediary server to relay data from client devices to the central server and an HTML5 page that uses the offline storage API so that support agents can access data offline.


Central database Online (datacenter)
HTTP Post Request syncs When connectivity available for given location ids
Intermediary server
Intermediary Server
Support Agent HTML5 Client downloads data For given location ids from master or intermediary server
Support Agent
Client Device
Client App (mini c++ web server) collects http requests from local content / applications and waits to transmit when a server is available

Task Breakdown

Central Server

1. Make a java application that will accept HTTP Post requests from intermediary servers accepting only specified IP addresses.
2. Design table /database structure

Intermediary Server

1. Make a small Java application that will:
    1. Accept HTTP Post requests from known devices and support agents to update data
    2. Pass on data to the central server when connectivity is available and record for each data row the data of transmission / status.
    3. Allow authorized support agents via a web interface to request a .zip file with the data for a certain data range.
2. The application must use a tiny Java Web server jar, not J2EE technologies
3. Use SQLite for database
4. Responses to clients should be in the form of XML / response codes. Remember: the Support agent will have it's own HTML5 Javascript app

Support Agent

1. Make an HTML5 HTML / Javascript app that will
    1. Connect to the intermediary server with a username and password and access data for given location Ids (AJAX → HTML5 offline storage)
    2. Allow the support agent to update device records for issuing a device to a user, conducting maintenance or support requests. Does not have to view data that the device has reported.
    3. Sync data back to the intermediary server when available.

Client Application

1. Write an daemon compiled to native code (e.g. C++) that will accept http post requests from the local system only (and store data using sqlite or similar) and then transmit to the intermediary server when it's available.