May 04, 2017

A novel method to discover datacenter hardware instantly using software defined grid model

Umesh M L
Hewlett Packard Enterprise

Naveena Kedlaya
Hewlett Packard Enterprise

A novel method to discover datacenter hardware instantly using software defined grid model

Abstract:-

Hardware (Servers, Storage, Switches) in Research & Development (R&D) datacenters may be used for ongoing research, development, testing, build, qualification and certification, performance, regression testing, resolving customer issues, and the like. Engineers may move the hardware frequently without involving the IT department to meet project requirements and may shutdown the hardware whenever not in use. Currently, there are no solutions available to locate the hardware when it is powered off. It may be useful to detect datacenter hardware instantly, even if the hardware is shutdown, powered off, or exists in an isolated private network. Further, a reporting module capable of reporting hardware location, serial number, configuration details, and the like may also be useful.

Prior Solutions:-

RFID

RFID solutions are comprised of RFID tags, which are fixed to every hardware in the datacenter. An asset management team reads these tags periodically using barcode readers and download the hardware information into a database. Potential drawback to this approach include:

1. An inability to provide a location of the hardware if there is hardware movement;
2. Inconsistent hardware data due to RFID tags peeling off from the hardware over a period of time.

Manual Tracking

A periodic, manual audit of datacenter hardware to maintain inventory in a spreadsheet may be undertaken, but such an audit is a laborious, manual, and time consuming process that is prone to human errors.

Monitoring tools

Available network and system management tools typically expect hardware to be accessible over a network but these tools do not provide whereabouts of the hardware in the datacenter.

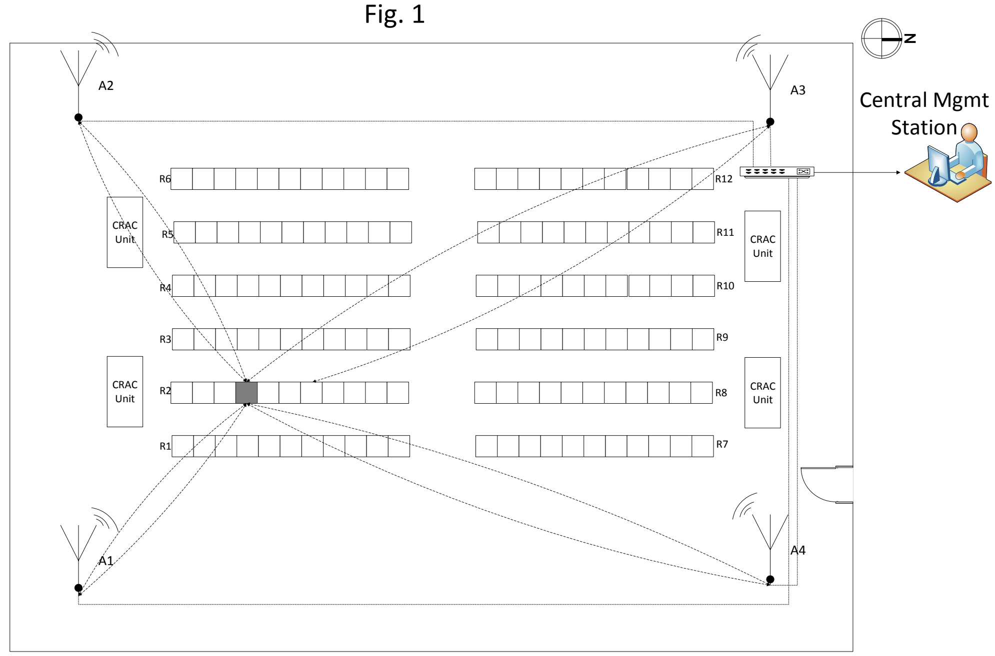
Invention Description:-

Our approach is broadly divided into the following three steps.

Step 1:-

We propose embedding a wireless receiver on the motherboard. These receivers have electronic circuits comprising of Accelerometer, Gyroscope and Magnetometer for determining the specific location, orientations and direction during the hardware movement. The adoption of wireless technology allows coverage of a wider range in the datacenter.

Fig. 1



Legend
MicroController to management switch
Round trip path between receiver and transmitter
Connection to Central Management Station

Physical location of the hardware to be tracked
R1 to R12 – Rows of racks
A1, A2, A3, A4 – Antennas
hosting wireless transmitters and MicroControllers

Schematic diagram of physical layout of the datacenter

Ceiling/floor antennas may be strategically planted in the datacenter to cover south, north, east and west directions, depending on datacenter floor design. For instance, four ceiling/floor antennas may be placed as shown in FIG. 1 above. An increase in the number of antennas may improve accuracy in locating the hardware in the datacenter. These antennas hosts wireless transmitter and microcontroller kits. The microcontroller kits (analog to digital converter) convert and transfer wireless signal data to a central management station via the local area network.

Step 2:-
The distance between a wireless transmitter and wireless receiver may be measured based on either wireless signal strength or based on a round-trip time. Our solution uses a round-trip time measurement to determine the distance between two wireless devices to minimize interference impacts within the datacenter.

We use the signal strength between each transmitter and receiver to choose the best possible transmitter. The reason for this is that the farther the transmitter antenna from the receiver, the more erroneous data that is recorded by the microcontroller kit.

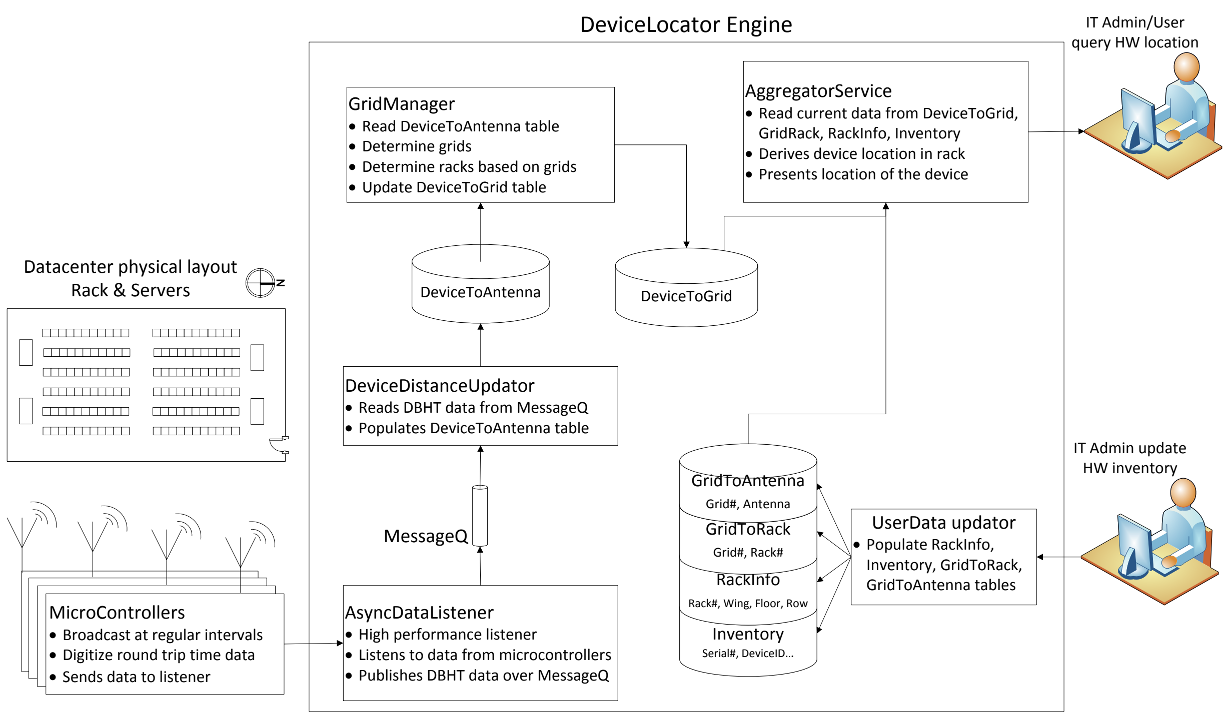
Each wireless transmitter antenna broadcasts the signal at regular intervals (e.g., once in 15 minutes) in a specific frequency. The broadcast is acknowledged by individual wireless devices embedded in the hardware. The transmitter antenna hosts a microcontroller kit that is responsible for converting analog data to digital data [distance between hardware and transmitter (DBHT)] and feeding this DBHT data to a backend DeviceLocator engine running on central management station.

Our solution also provides intelligent hardware tracking, which allows for a determination of the exact location of the hardware in the datacenter during hardware movement and alerts the IT department. During the server movement, the sensors in the receiver (described in Step1) become active. These sensors track movement periodically and store the information in a local flash memory. When the server is mounted in a rack at destination and power restored, this data is transferred to the central management station for further processing.

Step 3:-

The microcontroller converts the analog data to digital, received from the transmitter and sends it to an in-house designed backend DeviceLocator engine hosted on a central management station. This engine is comprised of different components and database tables to derive the potential physical location of the hardware. FIG. 2, below, depicts a proposed architecture.

Fig. 2



Architecture

3

AsyncDataListener: This component receives a continuous stream of DBHT data from each antenna and forwards it to *DeviceDistanceUpdator* through MessageQ. The data processing is delegated to other components in the system.

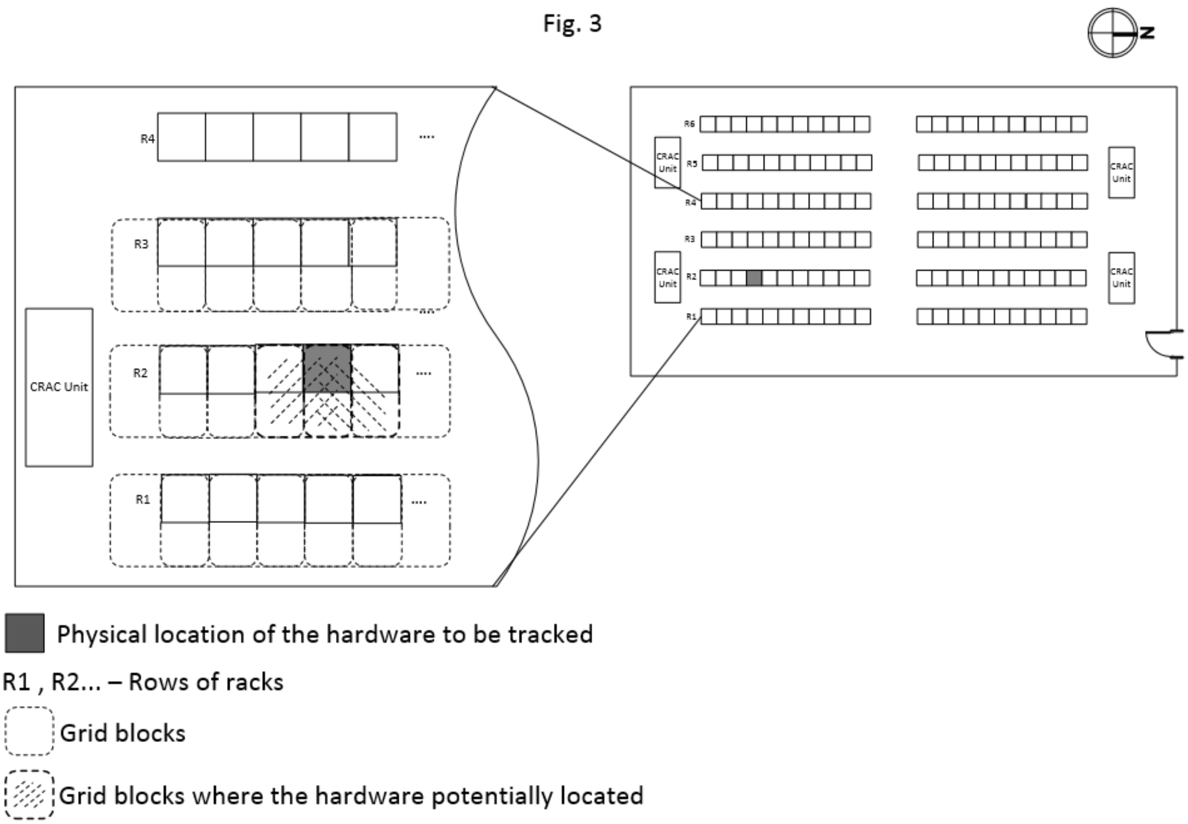
DeviceDistanceUpdator: This component receives the DBHT data from *AsyncDataListener* and updates *DeviceToAntenna* tables in batch mode.

DeviceToAntenna Table:
This table stores the real time data of DBHT and is updated periodically, for instance once every 15 minutes, based on the beacon sent from the transmitter.

Device id	A1	A2	A3	A4

GridManager: A grid model minimizes errors in hardware detection in a datacenter. One proposal uses grid blocks of 4 X 4 sq. ft. in the datacenter floor map, overlapped as shown in the below schematic diagram (FIG. 3). Larger or smaller sized grid blocks may also be used. The grids are marked with dotted lines and they effectively cover the entire datacenter rack placement. Based on signal strength, *GridManager* considers the respective transmitter data (DBHT) for processing and discards other values in the table. Based on this, *GridManager* determines the grid of the device fetched from *DeviceToAntenna* and *GridToAntenna* table and overlaid information is updated in *DeviceToGrid* table

Fig. 3



Grid Model Overlapping Datacenter Layout

GridToAntenna Table:
A mapping of distance between grid and antenna is stored in this table. The location of each grid can be calculated based on the Pythagoras theorem for a given antenna.

Grid#	A1	A2	A3	A4

DeviceToGrid table:
As described above, overlaid information is maintained in this table.

Device id	Grid#1	Grid#2

AggregatorService: With this module, users can get a location of the hardware and its configuration details. This module performs a look up in *DeviceToGrid* table and finds the grid number of the hardware. It then overlays grid information on the *GridToRack* table to determine the potential rack location. Then it does a lookup in *RackInfo* and *Inventory* tables to provide associated information such as the Floor, wing, Bond number, etc. of the hardware.

5

GridToRack Table:
Contains an association between the grid and the rack. This is captured in the *GridInfo* table, as shown below.

GridNumber	Rack#1	Rack#2

Inventory Table:
Inventory table is populated by the IT team when the new device arrives at the datacenter. Before the server is provisioned, a wireless receiver is attached. A sample table containing important data is depicted below.

Serial No	WIress ID	Bond #	HW Type	Serial #	Model	CPUs	Memory	Storage	Owner

RackInfo Table:
All the racks in a datacenter are inventoried. Details such as wing, floor, row number, and the like, help inform a user about the location of the device.

Rack No	Wing	Floor	Row

UserDataUpdator: Using this module, end users can update static information such as inventory, rack and grid information and grid to antenna distance.

Advantages:-

1. The proposed solution can be miniaturized to fit into a chip or can be integrated into a motherboard or management processor. This solution can also be made available via Peripheral Component Interconnect Express (PCIe or PCI-E) card or USB devices.
2. The proposed solution can be used in Machine Reservation Tool (MRT).

Disclosed by Umesh M. L. and Naveena Kedlaya, Hewlett Packard Enterprise