



CMUG – FEBRUARY 2018

Marimba by HARMAN

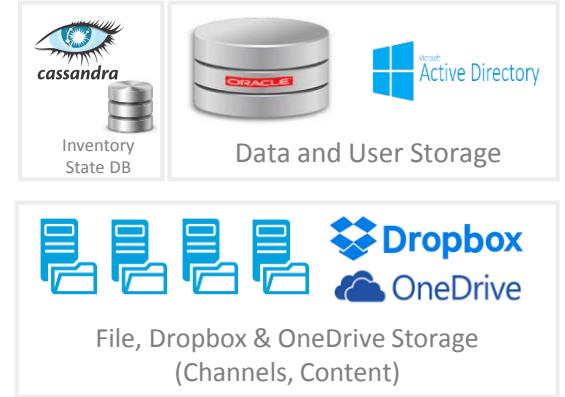
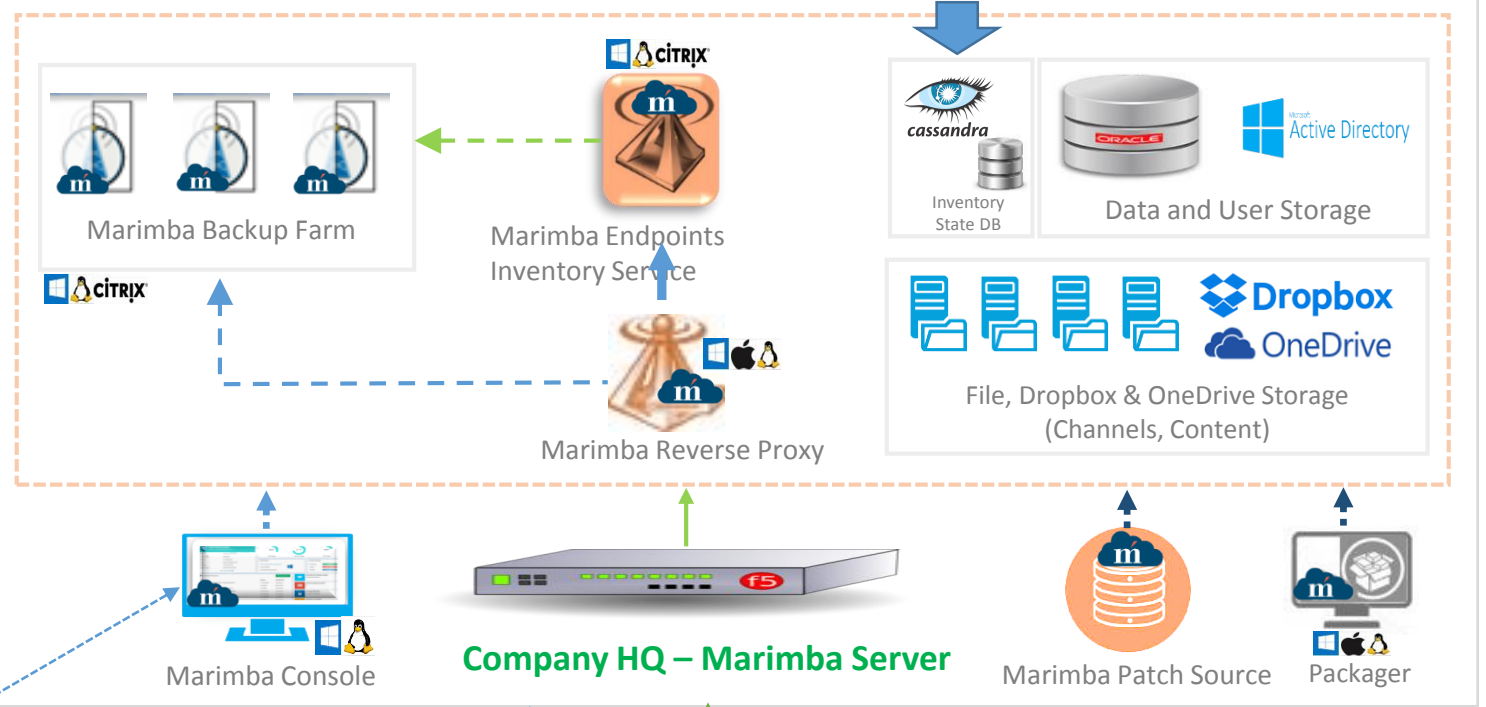
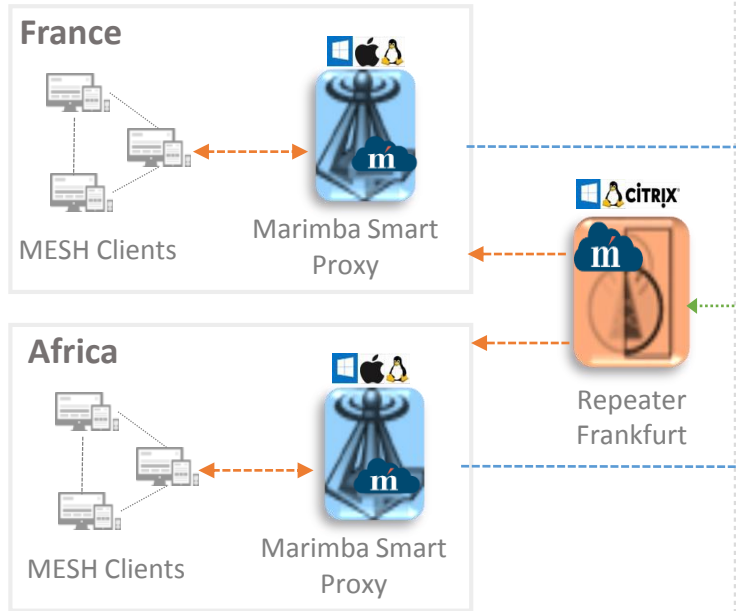
Marimba Inventory Redesign

Inventory Redesign for easy reporting.

- Checksums out of business
- The new model comes with a much more easy way of tracking the continuity of scan reports in the whole reporting process (by using a SEQ_ID that is now generated and included as part of each scan report).
- Performance bottlenecks caused by checksum mismatch at the plugins (resulting in full report insertion) is now prevented by an elegant, de-centralized tracking engine.
- The Plugin is accompanied by a tracking engine that ensures the reports sequence is maintained. Upon encountering a mismatch, it is auto-fixed online with the endpoint in the same conversation and the correct scan report is sent up in the processing chain.
- Assurance of differential reports is shot up to a higher extent with this State Database enabled Tracking Engine residing in each plugin (repeaters and mirrors both applicable)
- Enable it easily - It takes just one change in configuration to get the Tracking engine in action.

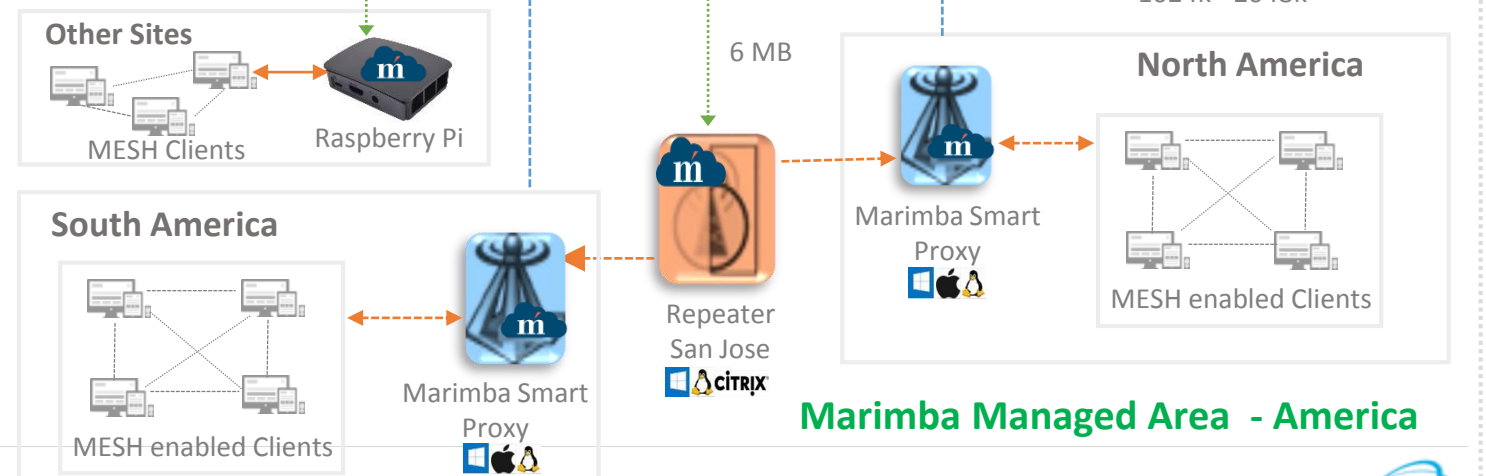
Result – Accuracy and Enhanced Performance

Marimba Managed Area - EMEA

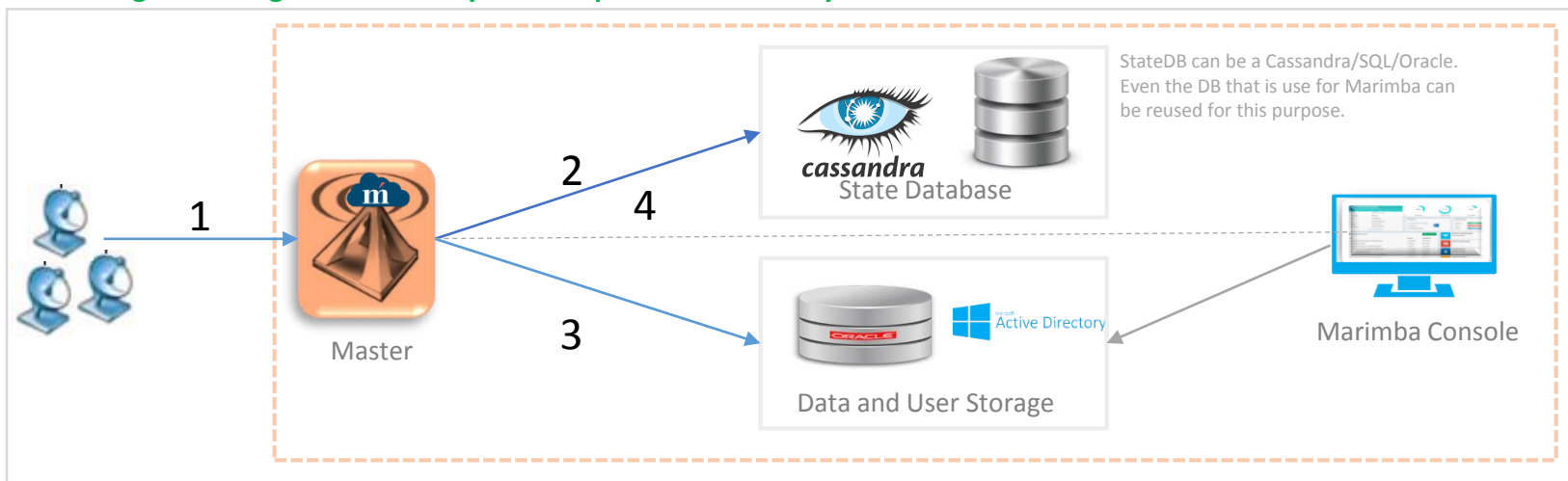


- Publish Link
- Request Link
- Download Link
- Replication or Redirection Link
- Request & Download Link
- Download through Mesh

Marimba can manage wide variety of platforms - Windows, Linux, Mac, Unix, Embedded Platforms, Windows Mobile Devices, IOS, Android, Embedded Linux, etc.

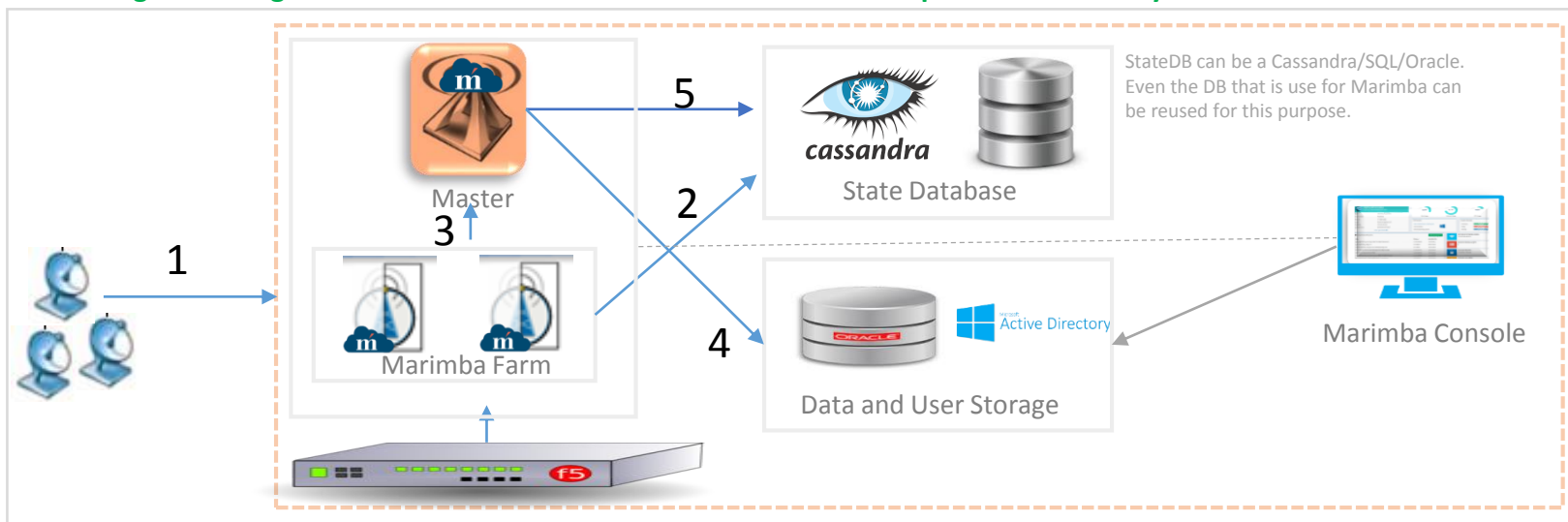


CASE: Plugin is configured to directly insert report into inventory database

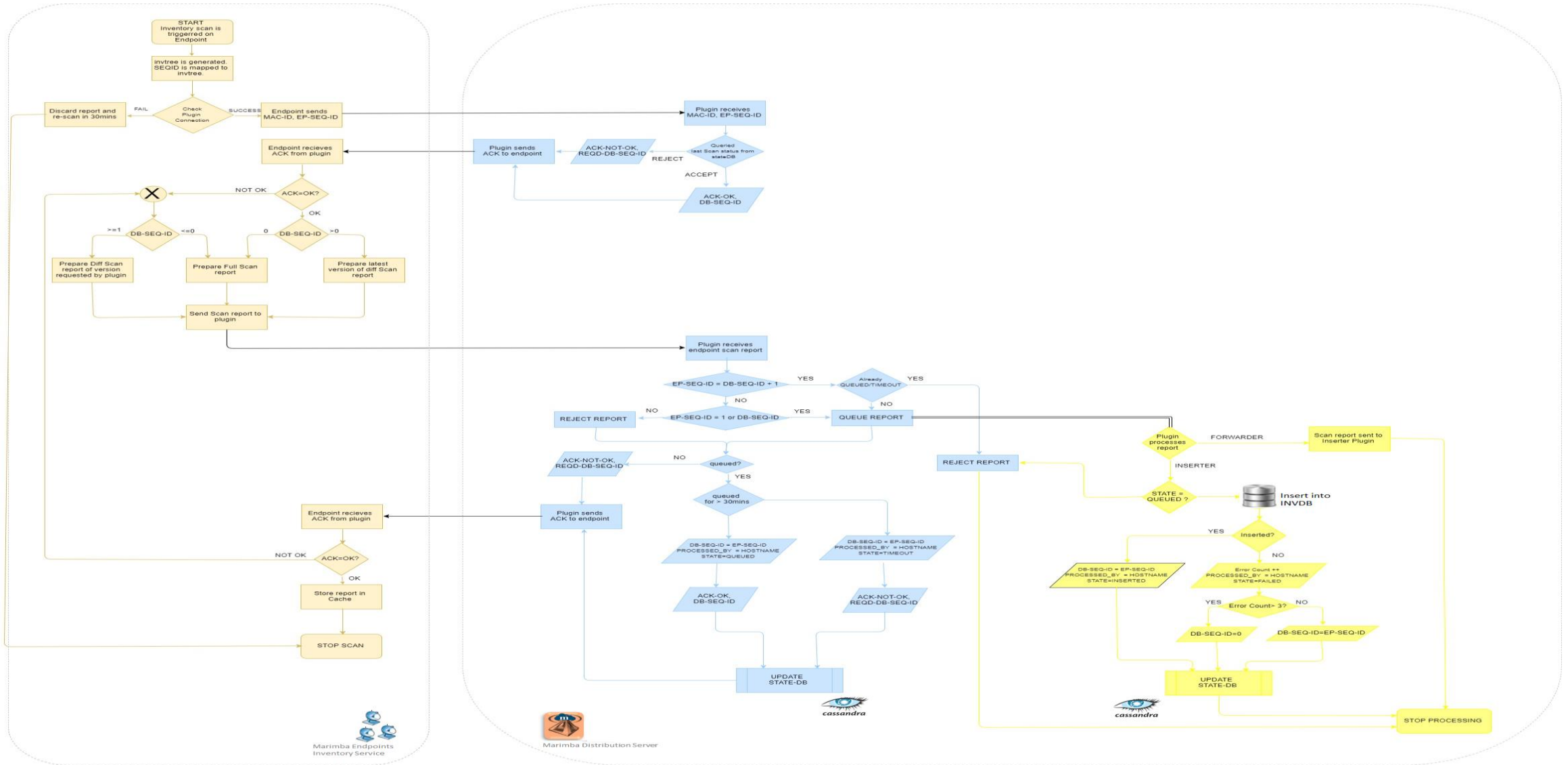


1. Endpoint scans and sends report to plugin based on acknowledgement received from the plugin
2. Plugin accepts/rejects report based on its query status from stateDB. It compares last scan status fetched from state DB for an endpoint with the version of report received from the Endpoint.
3. If plugin accepts report it queues and then inserts report into the inventory database
4. Plugin updates stateDB with latest status

CASE: Plugin is configured to forward to Inserter and Inserter inserts report into inventory database



1. Endpoint scans and sends report to plugin based on acknowledgement received from the plugin
2. Plugin accepts/rejects report based on its query status from stateDB. It compares last scan status fetched from state DB for an endpoint with the version of report received from the Endpoint.
3. If plugin accepts report it queues and then forwards it to the inserter
4. Inserter inserts the report into the inventory database
5. Inserter Plugin updates stateDB with latest status



Advantages

- Tracking is also done at the endpoint for easy resolution of conflicts in continuity of reports sent to the plugin.
- Assurance of differential reports is shot up to a higher extent with this State Database enabled Tracking Engine residing in each plugin (repeaters and mirrors both applicable)
- A query library is also available, which is built around the state database, helps administrators to track the time and performance stats of the flow of scan reports up in the processing chain.
- Error detection and correction both taken care of in the initial stage of the reporting chain.
- Distributed state database enables high performant error corrections, as the plugin detects the continuity of report relatively quicker.
- Bulk Checksum sync in repeaters and mirrors is out of scene now. This reduces load on the network on the repeaters and saves database roundtrips in the mirrors. Especially for geographically distributed repeater / mirror farms this will be a nice to have feature.
- Checksums are not sent along with the scan report (in the request headers) from the endpoints either. This reduces data being sent over the network.

Thank You

Send queries to Nitish.Shrivastava@harman.com

