

DECOUPLED SUPPORT

NEXT GENERATION DEVICE HEALTHCARE

Synopsis

The world is moving onto mobile computing devices. These present all sorts of support challenges and the most significant ones are addressed in Device HealthCare by Decoupled Support.

Precedent

Since mobile computing is a relatively new and rapidly evolving area, there is not much historical precedent for supporting systems that are intermittently connected to the support resource. The best examples are probably found in deep space exploration, where communications is low-bandwidth and intermittent, and software problems can disable an entire expensive mission.

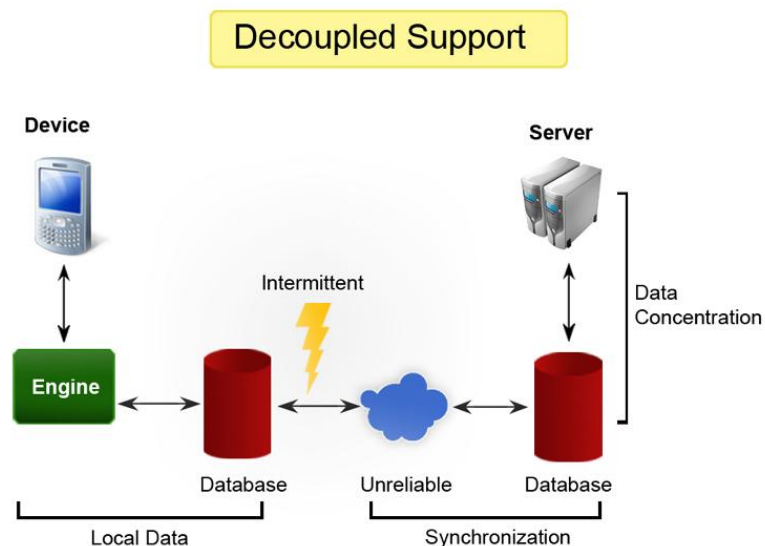
In this environment, the software is designed to be largely independent. This sort of design involves the uniform application of some basic principles:

- redundancy
- ability to detect catastrophic failure and reset to known initial state
- ability to use "dead reckoning" mechanisms to recover communications after a restart
- use of protocols that can recover from any failures mid-stream
- ability to download, verify, and process software updates with no unrecoverable failure modes
- ability to continue the mission even with arbitrarily long (or even permanent) communication failures

The Device HealthCare system architecture has many of the same characteristics, which give it true Decoupled Support.

Detail

As described in Distributed Support, DARTs (Diagnosis, Analysis, and Resolution Tools) locally implement solutions for support issues. This is a fundamental basis for Decoupled Support, because it means that the DARTs can manage support on a device for an arbitrarily long time without any external connection.



The server in the Device HealthCare paradigm is not primarily a control mechanism. The server's main functions are to act in the following roles:

- data concentrator
- convenient point for doing analysis to improve issue resolution
- distribution of updates
- distribution of configuration information
- escalation of general issues for manual attention

However, none of these functions are necessary, or even important, for device support. In addition, each function is also implemented locally and cached on the device, and the cache can be synchronized with the server if it ever becomes available.

The same databases, with the same schemas, exist on both the server and the local client. The local client implements them with a fast and efficient lightweight database engine, since it does not need to service multiple database clients. In this way, the devices cooperate with each other to manage distribution among them.

This means that the devices can operate with all of the server functions even if they never have access to the server, as long as they have intermittent access to each other, and at least one device has intermittent access to the server.

Benefit

The biggest benefit of Decoupled Support for a support organization is the strong support of mobile devices. There are other reasons why Device HealthCare is particularly well suited for mobile device support, but Decoupled Support is the most important one.

The other major benefit is the freedom from a critical central resource. Even if a server is down in the Device HealthCare paradigm, there is no serious service degradation. The devices go on working by themselves, and when the server becomes available again, the synchronization catches up and things continue normally.

Uniqueness

This is very simple. For any other offering, ask to see how it can run and perform all its functions without ever having a connection to a server. If it can't, then it doesn't have Decoupled Support, and it will always have trouble supporting truly mobile devices