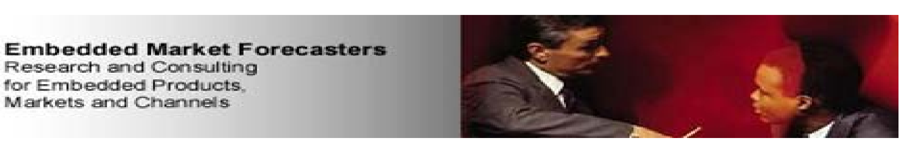
**EmbeddedMarket**
**forecasters**

# System Upgrades: The Challenge to IoT and Embedded Deployed Developments: Looking at Storage Technology


**Jerry Krasner, Ph.D., Chief Analyst**
**Dolores A. Krasner, Senior Editor**


**September 2015**





**American Technology International, Inc.**

**www.embeddedforecast.com**

## Regarding the Data in this report

The data that is referred to in this report is *statistically accurate* and *authentic* and is based on:

- A statistically generated comprehensive and detailed survey of embedded developers and managers who reported on their design results (number of developers per project, vertical market of their design, time to market, percent of designs completed behind schedule or cancelled, closeness of final design outcomes to pre-design expectations, testing outcomes, etc.), the tools they used (development, modeling, Java, Eclipse, and other development tools), their choice of OS, IDE, communication middleware, processors used as well as where they go to learn about new products, tools and concepts.

- An EMF Dashboard – a unique tool that allows the user to simultaneously compare similar products (vendors can do competitive comparative analysis); that marketing executives can use for sales promo and strategic planning; that allows developers beginning a project to compare the experiences of hundreds of fellow developers that undertook similar projects to gain insights before making a commitment; and that allows CFOs and senior managers to look at what tools and processes resulted in the greatest cost savings.

For the interested reader, the following link demonstrates the power of the Dashboard and how we used it in developing the data that is presented herein:

http://www.embeddedforecast.com/EMF_DashboardIntro/EMF_DashboardIntro.html

## System Upgrades: The Challenge to IoT and Embedded Deployed Developments: Looking at Storage Technology

**Jerry Krasner, Ph.D., MBA**

As part of the comprehensive 2015 EMF Survey of Embedded Developers (1334 respondents) engineers and managers were asked "Once deployed, how often is your current embedded project likely to be updated in the field?" The survey was filtered on this question as it was answered by developers that indicated that they were using Storage technology (capturing, collecting or creating data) as part of their embedded developments.

The following Table presents the results to this question for general embedded developments as well as for IoT developments.

| | Storage Industry | Storage IoT |
|---|---|---|
| **Never** | 23.8% | 5.7% |
| **Yearly** | 42.9% | 42.9% |
| **Quarterly** | 21.9% | 28.6% |
| **Monthly** | 4.8% | 8.6% |
| **More often than monthly** | 6.7% | 14.3% |

It is clear, but not unexpected, that IoT developments have substantially more upgrade requirements than those of the broader embedded industry

Developers were asked the follow on question "What are the cost implications of losing or corrupting the data stored in your current embedded project?" The results are presented in the following Table.

| | Storage Industry | Storage IoT |
|---|---|---|
| **Little or no cost** | 23.3% | 12.1% |
| **Minimal but acceptable cost** | 35.9% | 39.4% |
| **Major cost but not mission disabling** | 21.4% | 27.3% |
| **Mission critical** | 19.4% | 21.2% |

It is clear that the cost associated with data loss due to upgrades is greater for IoT developments than for similar embedded developments.

System updates are a key point of vulnerability for embedded systems. If not handled properly, a power loss or other unexpected shutdown can disable the entire system.

So what can developers do to protect themselves?

EMF turned to Datalight for their many decades of expertise in data storage and retrieval. In particular they explained to EMF how their Reliance Family of file systems is used to prevent data loss during power failures as well as during system upgrades.

EMF is not advocating any storage vendor herein, but feels that the Datalight solution is instructive for developers to learn from when designing systems that will require upgrading.

One major advantage of the Reliance family of file systems is the nature of their Dynamic Transaction Point™ technology. Writes and file updates occur in the working state of the media, and only following a successful transaction point are these updates designated as part of the “known good” state. This means that any data previously stored on the media will be intact in the event of an unexpected shutdown. The timing of a transaction point can be controlled by an application, and can even be suspended for a time when necessary. These key features allow a seamless system update to be completely reliable without having to write complex logic within the application software to account for possible interruptions.

What happens behind the scenes is this. Through a simple API, the application saves the existing transaction mask (a series of flags indicating which system transaction points are enabled) and then clears that mask. A manual transaction point is issued, which performs a final empty of the cache and establishes the known good state – in some ways, this is similar to the “checkpoint” done by a Microsoft Windows desktop. Any power interruption after this point will quickly boot the system to this state with no existing files lost or damaged.

New system or application files can then be installed and perhaps other files modified or deleted. Verification routines can also be run to validate the new system state – there is no rush to finish the process before a bored user hits the power button! Once the new system is ready to go, a manual transaction point is performed, which commits all the changes. Even this process can be interrupted by a power drop – until the transaction point is successfully completed, the system will always revert to the known good state.

The final step is to restore the previous transaction mask, allowing the device to operate as it was previously designed. Through one update or many, reliability is completely guaranteed.

EMF data clearly shows the frequency of systems upgrades and the potential associated cost. EMF believes that developers should take upgrades as a requirement when undertaking their designs.