Mercury Systems, Inc. Software Release Policy

Introduction

Mercury Systems defines three distinct levels of software release:

- **Major Releases** introduce entirely new software offerings or capabilities, organizational structures, and/or support for new or different hardware configurations. Major releases usually occur no more than once annually, and may occur less frequently. These are defined by a release change from, for example, 12.X to 13.X.
- **Minor Releases** introduce updates, minor features, and sometimes address new or different hardware configurations. These are defined by a release change from, for example, 12.1 to 12.2.
- **Point Releases** provide modifications to a minor release. These modifications may improve performance, add minor features, refine existing features, improve configuration support, and fix bugs. These are defined by a release change from, for example, 12.1.0 to 12.1.1.
- **Other Releases** can include updated releases of:
  - Products bundled within the Mercury software offerings
  - Products or software associated with Customer Special Requests (CSRs), which can include but not be limited to bug fixes, special pre-releases, custom enhancements, or contract-specific requirements
  - Pre-releases (also known as Early-Access or Development Kit releases) of product software, or Alpha and Beta test releases of new software products. These releases may be defined by the addition of an "a" or "b" to the versioning – for example, 12.a2 or 13.b1.

The release policy above applies to both LSP/BSP releases as well as higher level middleware or library releases. There is no requirement for release types to align in version – for example, an LSP/BSP may be at version 2.1 but the MultiCore Plus Middleware release that supports the hardware may be at version 13.2.

To enable customers to develop portable applications that can upgrade easily with new releases, Mercury is dedicated to maintaining stable APIs in all of Mercury's internally developed software products. This policy applies to Mercury internally maintained API (example: ICS) and does not imply implicit or explicit maintenance of APIs based on open standards that are generally available.

To accomplish this goal, Mercury Systems will:

- Support published application-programming interfaces (APIs) from release through the API life cycle (defined below).
- Selectively allow customers to receive legacy support for obsolete APIs.

API Life Cycle

The life cycle of an API passes through three stages:

- **Limited Availability Stage**
  - Available to customers participating in a Limited Availability Support Program
  - Includes new, custom, or funded legacy APIs
  - Subject to change
  - May be limited to specific hardware configurations
  - Requires no general customer notification of changes
  - New APIs may, but are not required to pass to the Production stage
- **Production Stage**
  - Shipped and documented with each release
  - Conforms to published internal or external specifications
  - Subject to change rarely and only with sufficient justification
  - Supported on new configurations
  - Could be modified or removed in the next major release
  - Works as specified on all target processors supported in the current release
- **Legacy Stage**
  - Remains in the code base, but shipped for legacy use only
  - May be removed from the code base in the next major or minor release or after 18 months have passed
  - Documented as a legacy function until removed from code base

3117.00E-0515-DS-software