Geomagic Plug-in SDK 4
Change Log

03/24/2006 – Release 4:

- **New Features:**
  - **Range Scan Widget:** Allows using a range device for collecting ordered data
  - **Photogrammetry Widget:** Allows using a range device for collecting ordered data with target registration performed on sticker targets.
  - **Mover:** Allows quick and easy construction of a device mover widget

  New Device Interfaces:
  - **RangeDeviceI:** Allows for the implementation of a range scanner.
  - **DeviceMover:** Allows for the implementation of a generic movement device.

  New Data Storage Classes:
  - **GMagicImage:** Container for an image.
  - **RangePointData:** Contains verticy coordinates as well as color and grid information.
  - **RangeScanData:** Contains a collection of RangePointData, Photogrammetry Targets, as well as live images.

02/23/2006 – Release 3:

- **New Features:**
  - **Probe Capture Widget:** Allows using a hard probe device for continuous capture of uncompensated points.
- **Improvements:**
  - **Hard Probing Feature/Datum Information Dialog:** Shows detailed information while collecting features/datums, such as fitting statistics, feature normals, and the hard probe's location.
  - **Laser Capture Clip Plane:** Allows definition of a plane below which no data will be scanned.
- **Miscellaneous:**
  - **Microsoft Visual Studio 8.0 Libraries:** Added libraries for use with Microsoft Visual Studio 8.0 (2005).
- **Interface Changes:**
  - **GMagicWidget:**
    - `id_()` and `localId_()` are const and no longer have const in their return types.
  - **ConnectorI:**
    - `isConnected()` is const and no longer has const in its return type.

        - `getUsageText()` is const.
    - **DeviceI:**
        - `getOperationText()` is const.
    - **TempDataSvc, PermDataSvc:**
        - Some methods are now const.
    - **PointData:**
        - Pointer constructors take pointers to const types.
        - Set methods have void return type.
        - `[]` accessor no longer has const in its return type.
    - **ScanData:**
        - `setPoints()` methods take pointers to const types.
        - `setPosition()`, `setToDirection()`, and `setUpDirection()` take const references.
        - `numPoints()` no longer has const in its return type.
    - **CloudData:**
        - Method Added: `bool deleteLastScanData();`
        - `numScans()` no longer has const in its return type.
    - **GMProbeFD:**
        - `getVersion()` has a return type.

12/01/2005 – Release 2 Final:

*** **Transformation File Reader**, Bug Fixes and GUI improvements, .NET Libraries ***

- **New Features:**
    - **Transformation File Reader:** Allows you to create and register a custom transformation file reader with the GMAlignment Widget. This allows hardware vendors to provide the user with the ability to load alignments from a proprietary file format.
- **Bug Fixes:**
    - **Hard Probing:** Fixed probe diameter compensation.
    - **GMagicUString:** Fixed constructor that caused random application crashes.
- **GUI Improvements:**
    - **Laser Capture:** Improved Scan Data grouping, and "Advanced Options" sensitivity.
- **Miscellaneous:**
    - **.NET Libraries:** Added libraries for use with Microsoft Visual Studio 7.0 (.NET)

11/14/2005 – Release 2 Beta 1:

***Changes to **ConnectorI, DeviceI, HardProbeSingleEventI, &**

LaserScanContinuousEventI Interfaces***

- **Interface Changes:**
  - **ConnectorI:**
    - Method Added: `GMagicStringGrp getUsageText();` This method allows you to specify a translatable string in the status bar of the Geomagic Application when the hardware is successfully connected. The method is called immediately after a call to `ConnectorI::connect()` returns true.
  - **DeviceI:**
    - Method Added: `GMagicStringGrp getOperationText();` This method allows you to specify a translatable string in the status bar of the Geomagic Application when the hardware is in the collect method. The method is called immediately before a call to `DeviceI::collect()`.
  - **HardProbeSingleEventI:**
    - Parameter Change:
      - `bool onMoved(const *ScanData);`
      - `bool onPointSampled(const *PointData);`
      - `bool onPointStop(const *PointData);`

      The parameter is now a Const Pointer to a ScanData / PointData object. The Plug-in Widget will take ownership of this object and will delete it when appropriate. Therefore, this object must be created with the `new` operator and you may not delete.
  - **LaserScanContinuousEventI:**
    - Parameter Change:
      - `bool onSampled(const *ScanData);`
      - `bool onMoved(const *ScanData);`

      The parameter is now a Const Pointer to a ScanData object. The Plug-in Widget will take ownership of this object and will delete it when appropriate. Therefore, this object must be created with the `new` operator and you may not delete.

2/7/2005 – Release 1:

Changes to **ConnectorI, DeviceI** Interfaces

- **Changes:**
  - **ConnectorI:**
    - The `ConnectorI::FunctionID` enumerations, and `ConnectorI::hasFunction,` and `ConnectorI::execFunction` methods have been removed. To support requests for more customizable User Interface, a Widget based framework was adopted. You will now be able to create simple GUI blocks called GMagicWidgets that will allow you to directly create any GUI interface and handle the GUI

interactions. See the documentation for more details.

- **Additions:**
  - **GMagicWidget Framework:**
    - Allows simple customizable GUI blocks to be added to a pre-packaged widget. This allows a developer to add any number of device dependant controls directly to the GUI. See the ExampleGPI project and the documentation more information.

12/14/2004 – BETA 4:

Changes to **LaserScanContinuousI** Interface

- **Changes:**
  - **LaserScanContinuousI:**
    - The `onStop()` method has been removed. It is sufficient to simply return from the `DeviceI::collect()`.
    - It is no longer necessary to precede calls to `onSampled()` with `onUnPaused()` because it is now called internally when an `onSampled()` event is received and capture has not yet started.
- **Bug Fixes:**
  - **Laser Capture:**
    - Stop the widget from recalculating the shading for scan passes created in a previous plug-in instance.
    - Updated the polygon display system to handle the special case of all points along a perfect line.
- **Additions:**
  - **ExampleGPI:**
    - The Laser Scanner portion of the DeviceI implementation now produces sample scan data.

12/8/2004 – BETA 3:

No Changes to Interfaces

- **Bug Fixes:**
  - **Alignment:**
    - In "Move Device," don't force the device to stop collecting points.
    - Make sure "3-2-1" is not sensitive if the hard probe connection fails.
  - **Hard Probing:**
    - Return true if data is successfully accepted from `onPointStop(const PointData & data)`.
  - **Laser Capture:**
    - Fixed a sensitivity issue when attempting to re-connect to the hardware if the connection is lost.
    - Call `disconnect()` when stopping the widget only if it is connected.

        - Refresh the scene even when Shading and Sampling are not checked.
    - **Laser Compare:**
        - Fixed a sensitivity issue when attempting to re-connect to the hardware if the connection is lost.
    - **Libs:**
        - Modified gmagicbaseRT.lib and gmagicutilsRT.lib to link properly when using MS Visual Studio 7 (.NET).

11/23/2004 – BETA 2:

Changes to **DeviceI**, **HardProbeSingleEventI** and **DeviceFactory** interfaces

- **New Features:**
    - **Alignment Widget:** Allows the user to establish a specific coordinate system that all incoming data will be applied to. Methods include: Default Arm Coordinates, Load from TFM file, Test / Feature Datum Object Transformation, with optional Hard Probe based alignment methods; 3-2-1 Plane Fitting and "Leap Frog" style Move Device.
- **Changes:**
    - **DeviceI:** A new method, `stopCollect()`, was added. This method should be implemented in such a way that if it is called while currently executing the `collect()` method, it should cause execution of the `collect()` method to complete. This allows data collection to be stopped programmatically.
    - **HardprobeSingleEventI:** A new callback method, `onMoved(const ScanData & sd)`, was added. This callback method should be called anytime the probe is moved but no data is actually being collected. It is used in the "Show Coordinates" functionality of the Alignment Widget. *(This is currently only necessary for Hard probe capable devices.)*
    - **Device Factory:** The Device Factory now requires a DeviceFactoryImpl subclass to return a pointer to a DeviceI implementation from within the inherited `get()` method. See the documentation and examplegpi for more information.

11/18/2004 – BETA 1:

Initial Beta Release

- **Features:**
    - **Laser Scanning Widget:** Allows user to collect data with a laser line based scanner. Includes options such as Virtual Camera, and Real-Time 3D Meshing.
    - **Laser Comparison Widget:** Allows user to compare laser line data as it is collected to a CAD Reference model. The Deviation texture map of the

CAD model is updated in real-time.

- **Hard Probe Widget:** Allows user to define Features and Datums by sampling points with a Hard Probe Device.