Eclipse ImageSignature Feature

ECLIPSE
DATA TECHNOLOGIES

Background:
Eclipse was the first in the industry to integrate checksum protection into the premastering and data verification workflow. Our ImageIntegrity technology was introduced more than eight years ago and has been part of every ImageEncoder system ever sold. ImageIntegrity is also a key component of the EclipseSuite software and our electronic file transfer products. Eclipse has long understood the advantages of checksum protection and we are the proven leader in data verification solutions.

Limitation of simple CRC's:
Traditional checksum calculations are simple. A checksum routine is performed on a data file and a CRC value is computed. If the data changes in any way, the checksum changes. Basic checksums work well to protect hard drive images, but they fail to be useful when trying to compare two things that are not exactly the same.

Variation between an image and a disc:
Most formats have data fields that are altered during mastering. CD may need to have post-gap added or a link block removed. DVD needs to be encrypted and have mastering instruction bits reset. An encoder has to manipulate the data in specific ways in order to make a conforming disc. A traditional checksum cannot be used to compare the original image to the replica because the data is not exactly the same. The CRC values cannot be expected to match.

ImageSignatures:
To get around this limitation, it became necessary to implement some additional logic into the calculation of a CRC value. ImageSignature is just that: An advanced checksum that takes into account the known and expected variations between a master image and a final replica. Since the master and the replica are expected to have the same Signature, it is possible to use this calculation for data verification and image identification. Although incredibly robust, the Eclipse Signature is small in size and is easy to manage. Signatures are human readable and can be input and stored in a number of flexible ways. Signature verification is done using ImageVerify allowing you to keep your process virtually unchanged. ImageVerify still provides full analysis of the replica and complies with all CSS licensing requirements. ImageSignature is a new feature free to all EclipseSuite 4.0 systems licensed with ImageIntegrity.

Benefits of ImageSignature verification:
Our customers have requested an advanced verification solution for two main reasons:

1. Efficiency is improved if the master is not required for data verification. This translates into less use of network resources and less time reading tapes**.**
2. The original master can be tightly controlled and does not have to be managed. This improves overall security concerns associated with today's manufacturing environment and makes for a simpler process.

Eclipse ImageSignature Feature

Definition of Signature:
A Signature is a small, but statistically unique number calculated across an image. The Signature must be the same for all representations of that image, regardless of:

- Source storage mode – indicates whether an image is pre-encrypted, to be encrypted, not to be encrypted, whether sectors for encryption have been pre-selected at authoring time or to be chosen by the encoder at mastering time, whether keys are present with the image or separate on a floppy disc
- Record size: common record sizes for an image are 2048, 2054, 2064, 2336, 2352
- Fully processed vs. user data– whether the sync, header, EDC, and ECC bytes are physically in each record (not necessarily calculated yet).
- Complete vs. Incomplete: when the record size includes space for the sync, header, EDC and ECC bytes, this flag indicates whether those bytes have been pre-calculated, or if there are just dummy "place-holder" bytes in the record, meaning this will be calculated at mastering.

Robust Algorithm:
The Eclipse Signature value is computed using the MD5 algorithm. MD5 is an advanced "message-digest" routine that creates a 128-bit "fingerprint" for each given data input. MD5 is a way to verify data integrity. MD5 is more reliable and secure than conventional checksum methods and is the foundation of secure banking transactions and government encryption.

Theoretically, MD5 will yield a false match 1 in 2^{128} times (10^{38}). This number is equivalent to the number of water molecules in 2.7 million Olympic size swimming pools. With each drop of water in each pool having billions of molecules in it. The statistical probability of a Signature failure is so low that it is hard to comprehend the odds.

DVD Calculation Methods:

- For added strength, the Signature calculation includes control.dat values for fixed fields such as layer number, translation, and sector address info. Control values that may be changed during mastering are omitted from the Signature.
- The Signature excludes zero filled data sectors to address padding variations.
- The calculation for each sector will include all user data and the sector number.
- For each VOB sector, the Signature calculation exclude:
    - CSS Keys: The same image created with different keys will have the same ImageSignature. CSS encryption will be validated by ImageVerify CSS during the verification process.
    - The Signature calculation will also exclude all specific bits in the user data that contain encryption flags. The same image, but with different sectors chosen for encryption, will still have the same signature.
- Copy Protections: Not currently supported.
- Type 2 SSCRST tapes are not supported.

Eclipse ImageSignature Feature

CD Calculation Methods:

- For all discs, the Signature calculation will include the UPC code, Table Of Contents, and all ISRC codes.
- The Leadout point will not be included to allow for postgap changes
- CD ROM Tracks:
  - For each sector: 2048 byte user data + Track # + Index # + ATIME
  - Zero filled sectors (tolerating postgap corrections), track descriptor blocks, and link blocks are excluded from the Signature calculation.
- Audio Tracks: not supported.

Are these Signatures the Same?

- Any Changes in User Data = Signatures Comparison Error.
- Region Code Changes Main Data = Signature Comparison Error.
- Padding removed = Signatures OK
- Control Data Changes: Book Type, Part #, Manufacturer = Signatures OK
- Control Data: Layer Number, Translation type = Signatures Comparison Error
- Different CSS Keys Used: Signatures OK
- Different sectors selected for encryption: Signatures OK
- PostGap Added: Signatures OK

Conclusion:

Adopting a Signature verification process can increase efficiency and security. Signature verification eliminates the need to read the original master for data verification purposes. Signatures offer robust protection and can be implemented with very little modification to an existing ImageVerify process. Please contact your Eclipse sales representative to discuss how Signatures might work for you.