

# Blade v1.10

- [Introduction](#)
- [SQLite Database Recovery](#)
- [Jump List Recovery](#)
- [Other New Features](#)
- [Change Log](#)

## Introduction

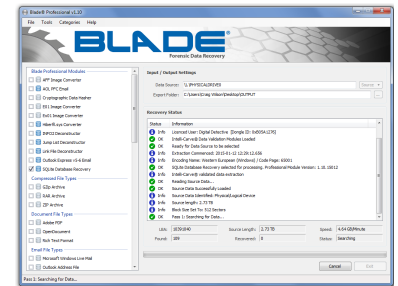
This release of Blade® was unfortunately delayed during the development of [NetAnalysis® v2](#) and [HstEx® v4](#). We have been working hard over the past few months and are pleased to announce the release of Blade v1.10. This release fixes a number of outstanding issues and adds some new features. We have also made some changes to increase performance and stability. To see a full list of the changes we have implemented, please review the following: [Change Log v1.10](#).

## SQLite Database Recovery

SQLite is a software library that implements a [self-contained](#), [serverless](#), [zero-configuration](#), [transactional](#) SQL database engine. It is the [most widely deployed](#) SQL database engine in the world and SQLite databases can be found on almost every digital device we may wish to examine. Our SQLite database recovery profile utilises Intelli-Carve® technology to find and extract SQLite databases. The recovery engine understands the structure of SQLite databases and can verify the integrity of the database during the recovery process.

As each database is recovered, it is checked for integrity. Any database which fails the integrity check is copied to a separate folder so that it may be manually checked at a later date. Any databases which are not fully recovered are also copied to a separate folder so they may be manually checked at a later date. All databases which pass integrity verification are copied to the main folder so that they may be examined first.

Name	Date modified	Type	Size
Partial Recovery	2015-01-13 16:38	File folder	
Recovered with Integrity Errors	2015-01-13 16:38	File folder	
PS-0000000304 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000000312 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000000328 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000000336 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000000344 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000000360 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	108,975 KB
PS-0000218312 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	84,871 KB
PS-0000390856 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	6,311 KB
PS-0000419152 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	113,522 KB
PS-0000646200 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	84,871 KB
PS-0000815944 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	1,322 KB
PS-0000819272 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000819280 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	74 KB
PS-0000819432 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000819448 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000819456 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000819464 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB
PS-0000819472 SO-000-recovered.sqlite	2015-01-13 16:36	SQLITE File	3 KB



The data validation log keeps track of which databases failed the integrity check and also logs the reasons for their failure.

## Jump List Recovery

Jump Lists are an interesting and common forensic artefact found when examining Microsoft Windows 7 or 8. They are a Taskbar feature that allow the user to quickly access recently accessed files and actions associated with a particular application.

Automatic Jump Lists (.automaticDestinations-ms files) are created by the operating system. These files are OLE Compound Files which contain in each stream a Windows Link File structure. There is also one special DestList stream which holds Most Recently Used (MRU) or Most Frequently Used (MFU) information for each of the Link File streams.

Our Jump List recovery profile deconstructs Automatic Jump Lists and for each Link File stream it writes out the recovered information to either CSV or Excel Specific CSV. The corresponding DestList information is prepended to each CSV Link File record. The user can also decide whether each Link File structure from the original source is also written out.

## Other New Features

We have changed the way Blade® searches for artefacts and have implemented parallel processing which allows us to use more CPU processor cores. This should considerably increase the searching performance when multiple data recovery profiles are selected. We have now added support for installing Blade® on Microsoft Windows 8. In relation to supported image types, we have now added native support for EnCase® 7 ex01 image files.

Another important change in this version is a major upgrade to the recovery engine. Previously, Blade® would struggle to deal with the recovery of millions of files; this is no longer the case.

## Change Log

The full change log can be found here: [Change Log v1.10](#)