What Makes Build-Time SBOMs Different from Other SBOM Generators?

Build-time SBOMs capture only the components and libraries in a build for an accurate picture of your software. This approach eliminates false positives and negatives and is ideal for real-time software and software written in C/C++.

	Build-Time SBOMs (RunSafe Identify)	Source Code-Based SBOMs	Binary-Based SBOMs
Analyzes software during the compilation process	✔	✘	✘
	Provides 100% visibility into components		
Reduces false negatives and positives	✔	✘	✘
	Delivers precise & complete SBOMs		
Accurately reports on dependencies	✔	✘	✘
Captures both static and dynamic libraries	✔	✘	✘
Ideal for complex codebases, particularly C/C++	✔	✘	✘
Reports on all opened files	✔	✘	✘
	Includes source, header & compiled files, libraries, applications		
Includes provenance information, like supplier, author, and license	✔	✔	✘
Delivers SBOM with all minimum NTIA fields	✔	✔	✘