

Requirements for a device-independent format:

1. Preserve the high-level intent or semantics of the Graphics calls (e.g. Java AWT Graphics) used by an application e.g, do not convert graphics calls to a bit image.
2. Simple things ought to be simple! Should not have to deal with undue complexity to print a simple page of text, for example.
3. Reduces client-side processing to map Graphics calls to device-independent format.
4. Enables utilization of the text/graphics capabilities of a target printer for optimal print quality and performance.
5. Encourages the development of printers that can directly consume the format
6. Provides a more efficient format over the network and in storage.
7. Maintains the semantics to make the device-independent format useful between applications, not just for print.
8. Supports, but is not necessarily limited to, Java 1.1.x AWT Graphics and Java 1.2 2D Graphics.
9. Streamable format from the client to the server i.e. allow a job to be sent via multiple sends so the client doesn't have to store the entire job before sending the job to the server.
10. Specify fonts by family name and style i.e. do not require (but don't necessarily preclude) the complete set of font metrics, glyphs etc to be carried in the format.
11. Support Unicode, DBCS and UDC (user-defined) characters.
12. Support standard image formats (e.g. GIF, JPEG) and compression algorithms for efficient storage of images.
13. Store multiple references to the same resource (e.g. image) rather than multiple copies of the resource.
14. Provide direct access to pages in a job to print a range of pages and to print the pages in reverse or n-up order.
15. List resources (e.g. fonts, image references) used in a job to enable the server to download resources in advance of printing the job.
16. Support versioning and upward compatibility.
17. Store data types in an operating system neutral manner.

18. Easily extensible to support future Graphics calls.
19. It would be good if a device-independent format could be processed in a linear fashion from beginning to end. Just as you want to stream on output to save memory, you want to process on the fly on receipt, particularly if you ever implement device-independent format in a printer.
20. Format can be saved and retrieved on demand for subsequent printing.
21. Format viewable via a browser/plugin.
22. If possible, the format should be an existing or emerging standard.
23. You should be able to go roundtrip from the application's Graphics calls to the format to the original Graphics calls without a loss of information or function.
24. Support job-level and page-level print options.