IBM

WebSphere Application Server for z/OS
# Summary of Enhancements

- **Based on Customer Feedback**
- **Many are Exclusive to WAS on z/OS**
- **Many are Related to Enhanced QoS**

Version 6.1

Version 6.1 marks a major milestone for WAS z/OS. It was with V6.1 that all the supported operating system platforms for WAS unified around a common and consistent set of standard specification support. In addition, the following enhancements were included:

- `z/OS` • IBM z/OS SDK 5, which provided: 64-bit SDK support; Large Page and Compressed References support (a performance enhancement); and z9 processor exploitation (another performance enhancement)
- `All` • WSADMIN AdminTask, which made changing the configured system and host names *much* easier, which in turn made testing Disaster Recovery (DR) easier and less prone to error.
- `z/OS` • Many z/OS DISPLAY commands, which provided much greater insight into operational status, and the ability to pause application server listener (PAUSELISTENERS) while leaving in-flight work processing to completion. These enhancements are documented in detail in the WP101138 Techdoc at `ibm.com/support/techdocs`.
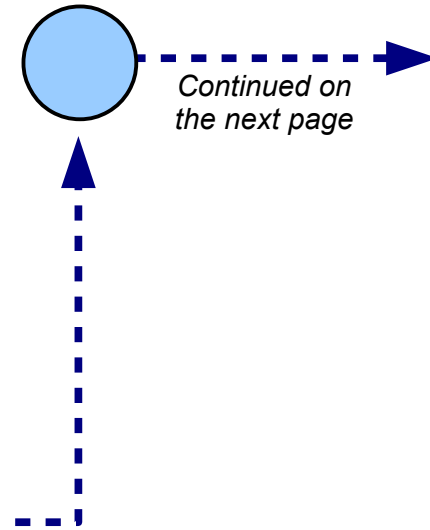
Version 6.0

Version 6.0 provided significant commonality for WAS across operating system platforms. Enhancements in this version included:

- `All` • Updates to standard specifications support
- `z/OS` • Ability to define a WAS z/OS cell to include platforms of different operating systems. This is referred to as a heterogeneous cell and was documented in detail in the WP100644 Techdoc at `ibm.com/support/techdocs`.

Version 5.x

Version 5 represented a departure from the V4 model and a movement towards commonality with WAS across operating system platforms that culminated in complete unity in V6.1. Enhancements to WAS z/OS in the 5.0 and 5.1 timeframe included:

- `z/OS` • ISPF panels to improve runtime creation process
- `z/OS` • Switch JVM from EBCDIC to ASCII
- `All` • Updates to standard specifications support
- `z/OS` • IBM z/OS Java SDK capable of offloading to zAAP speciality engines

Version 4

First Java EE version of WAS on z/OS. Had several acknowledged drawbacks: required DB2 and LDAP for configuration; manually intensive runtime configuration process; different architecture and specification support from WAS on distributed; JVM operated in EBCDIC code page.

User experiences based only on Version 4 are understandably less positive than user experiences based on later versions of WebSphere Application Server on z/OS.

IBM

WebSphere Application Server for z/OS
Summary of Enhancements

- Based on Customer Feedback
- Many are Exclusive to WAS on z/OS
- Many are Related to Enhanced QoS

Version 8.5

The latest version of WebSphere Application Server ... enhancements include:

- **All** • A new server runtime model called "Liberty Profile" that provides a lightweight and fast server with dynamic and composable functionality. Available on all platforms, **z/OS** and extended with z/OS functions. See the WP102110 Techdoc for more on Liberty for z/OS.

- **All** • The inclusion of the IBM WebSphere Compute Grid function for processing Java batch within the WAS runtime container model. Compute Grid z/OS with **z/OS** z/OS exclusive functions. See WP101783 for more on WebSphere Java Batch.

Version 8

Following on Version 7, WAS z/OS Version 8 provided the following enhancements:

- **All** • Fairly significant updates to the standard specification support (see this InfoCenter article for a listing)

- **z/OS** • IBM z/OS SDK 6.0.1, whichprovided: exploitation of the z196 processor; enhanced JZOS and z/OS security; and an improved internal virtual machine and just-in-time compiler that considerably enhanced performance)

- **z/OS** • Additional enhancements to the WebSphere Optimized Local Adapters support (see "History of Updates" at WP101490 for details)

- **All** • A mechanism for defining an "alternate JNDI" for enhanced data access availability. Available on all platforms, this provided a way for the WAS container to automatically failover to a defined alternate JNDI in the event of `getConnection()` failure, and automatic failback when the primary data resource had recovered. **z/OS** z/OS support extended with MODIFY to failover and failback to support *planned* outages for maintenance and update.

- **z/OS** • A way to drive WAS behavior -- such as timeout values, CPU used limits, SMF record, tracing activity -- down to the individual request level. This allows different applications to be deployed to the same application server while enjoying differential runtime behaviors.

Version 7

This version may be thought of as the version where WAS on z/OS really took flight. This is the version where WAS z/OS took specific advantage of several key functions of the z/OS operating system. We called these new features "zDiff," meaning they differentiated WAS z/OS from WAS on other platforms while maintaining the common and consistent standard specification support across the entire platform family.

Key new features and functions in WAS z/OS Version 7 included:

- **All** • Updates to standard specifications support

- **z/OS** • Elimination of the ISPF customizaton panels in favor of a graphical workstation tool that integrated seamlessly with the planning spreadsheets provided at PRS3341 at `ibm.com/support/techdocs`.

- **z/OS** • IBM z/OS SDK 6, which provided exploitation of the z10 processors

- **z/OS** • WebSphere Optimized Local Adapters (WOLA), which provided a high-throughput, low-latency, bi-directional mechanism for exchanging messages between a WAS z/OS server and another address space on the LPAR, such as CICS, IMS, Batch, USS processes or ALCS. See Techdoc WP101490 for more on WOLA, as well as this IBM developerWorks article authored by the lead architect of WOLA.

- **z/OS** • A new SMF record architected from the ground up based on customer input. This new SMF 120.9 record provides substantial information about requests received by the application server. The overhead of capturing the records was found to be significantly lower than the previous SMF records. The SMF recording was configurable with static definitions or dynamic enablement with MODIFY. Techdoc WP101342 provides a detailed description of this new SMF record.

- **z/OS** • A new mechanism designed to help avoid or delay EC3 abends of the WAS z/OS servant region when a Java thread was marked hung. This provided a way to avoid servant outages for cases where request timeout values were exceeded due to occasional long processing times.
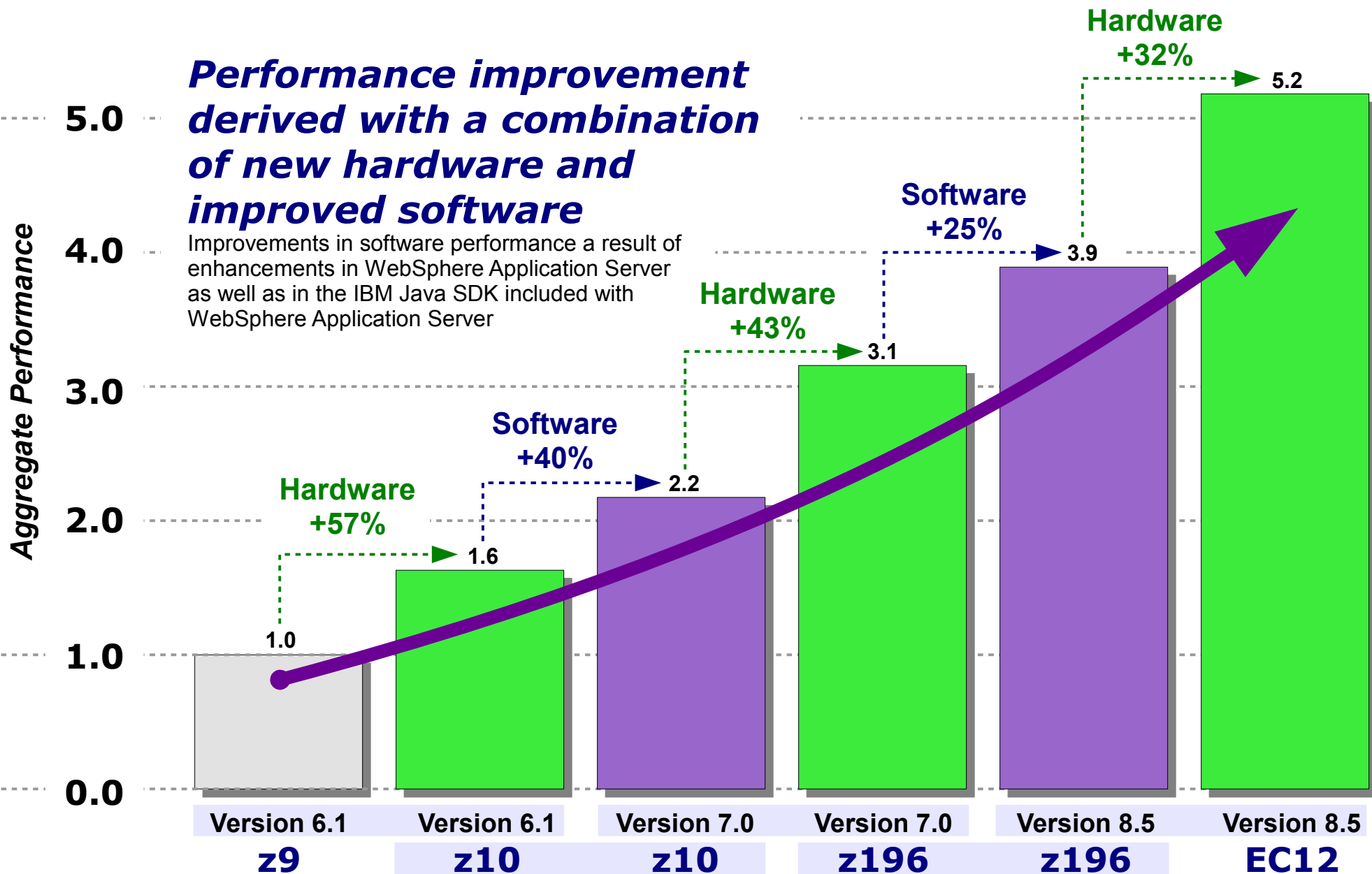
Continued from the previous page

Performance improvement derived with a combination of new hardware and improved software

Improvements in software performance a result of enhancements in WebSphere Application Server as well as in the IBM Java SDK included with WebSphere Application Server

Performance measurement conducted in a controlled environment under specific conditions. Your results may vary based on many factors that influence overall system performance. Results here are not a promise of similar results. Results here are for traditional WebSphere Application Server z/OS and not the new Liberty Profile server. Version 8 not shown here because Version 8 and Version 8.5 are roughly equivalent in performance profile. Workload based on "DayTrader" benchmark sample. Details on the DayTrader application may be found at this URL: https://cwiki.apache.org/GMOxDOC20/daytrader.html