| | ARMv8 | MIPS64 R6 | Power v3.0 | RV64G | SPARCv9 |
|---|---|---|---|---|---|
| Original date (base ISA) | 1986 | 1986 | 1990 | 2016 | 1987 |
| Date of this ISA | 2011 | 2014 | 2013 | 2016 | 2008 |
| Instruction size (bits) | 32 | 32 | 32 | 32 | 32 |
| Address space (size, model) | 64 bits (flat) | 64 bits (flat) | 64 bits (flat) | 64 bits (flat) | 64 bits (flat) |
| Data alignment | Aligned preferred | Aligned preferred | Unaligned | Aligned preferred | Aligned |
| Data addressing modes | 8 (including scaled, pre/post increment) | 1 (+1 for FP only) | 4 | 1 | 2 |
| Integer registers (number, model, size) | 31 GPR x 64, plus stack pointer | 31 GPR x 64 bits | 31 GPR x 64 bits | 31 GPR x 64 bits | 31 GPR x 64 bits |
| Separate floating-point registers | 32x32 or 32x64 bits | 32 x 32 or 32 x 64 bits | 32 x 32 or 32 x 64 bits | 32 x 32 or 32 x 64 bits | 32 x 32 or 32 x 64 bits |
| Floating-point format | IEEE 754 single, double | IEEE 754 single, double | IEEE 754 single, double | IEEE 754 single, double | IEEE 754 single, double |

**FIGURE E.1 Summary of the most recent version of five architectures for desktop, server, and PMD use (all had earlier versions).** Except for the number of data address modes and some instruction set details, the integer instruction sets of these architectures are very similar. Contrast this with Figure E.29. In ARMv8, register 31 is a 0 (like register 0 in the other architectures), but when it is used in a load or store, it is the current stack pointer, a special purpose register. We can either think of SP-based addressing as a different mode (which is how the assembly mnemonics operate) or as simply a register + offset addressing mode (which is how the instruction is encoded).