

Aspect	<code>RequestDispatcher.forward()</code>	<code>response.sendRedirect()</code>
<b>Execution Type</b>	Server-side internal dispatch	Client-side redirection
<b>Where it happens</b>	Inside the servlet container (server)	Browser makes a new request
<b>Request Object</b>	Same request object is reused	New request object is created
<b>Response Object</b>	Same response object	New response cycle
<b>URL in Browser</b>	Does NOT change	Changes to new URL
<b>Request Attributes</b>	Preserved (can pass data using <code>request.setAttribute()</code> )	Lost (new request, attributes not available)
<b>Session Attributes</b>	Available	Available (session survives both)
<b>Performance</b>	Faster (no extra network round trip)	Slower (extra HTTP round trip)
<b>HTTP Status Code</b>	200 (normal response)	302 (temporary redirect by default)
<b>Can Redirect Outside Application?</b>	No (only within same web app)	Yes (can redirect to external sites)
<b>Browser Involvement</b>	Browser unaware of forward	Browser fully involved
<b>Use Case</b>	Passing control between servlets/JSP in same application	Redirecting after form submission (PRG pattern)
<b>Security</b>	More secure (internal, user cannot see target resource path change)	Target URL visible to user
<b>Form Resubmission Risk</b>	Refresh may resubmit form	Prevents duplicate form submission (when used after POST)