

Server Side Rendering

```
graph LR; Request[ ] --> WebServer[Web Server]; WebServer --> Engines[Engines  
Servlet Engine / PHP / ASP.NET  
NodeJS / ExpressJS]; Engines --> Cache[( )]; Cache --> Engines; Engines --> WebServer; Engines --> Presentation[Presentation HTML / PDF]; Presentation --> Engines;
```

The diagram illustrates the Server Side Rendering process. It begins with an incoming request (represented by a black arrow) that enters a **Web Server** (a vertical rectangle). From the Web Server, the request is passed to a box labeled **Engines**, which contains two categories of technologies: **Servlet Engine / PHP / ASP.NET** and **NodeJS / ExpressJS**. A green arrow points from the Engines box to a green-outlined cylinder representing a **Cache**, and another green arrow points back from the Cache to the Engines box. A blue arrow points from the Engines box back to the Web Server. Finally, a blue arrow points from the Engines box to a document icon labeled **Presentation HTML / PDF**. A feedback loop is shown with a black arrow pointing from the Presentation output back to the Engines box.

The diagram illustrates the Client side Rendering process. It starts with a request (indicated by a long arrow) entering a **Web Server**. The Web Server then interacts with a box containing **Engines** (Servlet Engine / PHP / ASP.NET) and **NodeJS / ExpressJS**. This box is connected to a **Database** (represented by a cylinder) via a **Data** flow. The Web Server outputs a **Representation** (XML / JSON) to a box labeled **Libraries/Frameworks to convert representation to presentation**. The Database also outputs a **Representation** (XML / JSON).