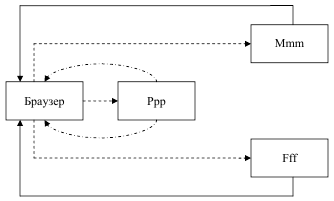
1. **Что такое переадресация?**

# **Переадресация**



Принципиальное отличие переадресации от переопределения запроса заключается в том, что передача управления от сервлета А к сервлету В осуществляется через посредника – клиента сервлета A.

Штриховыми линиями изображается вызов браузером сервлетов, штрихпунктирными – переадресация, а сплошными – движение информации. Браузер вызывает сервлет **Ppp**, в котором в зависимости от некоторого условия осуществляется переадресация на сервлет**Mmm**или сервлет**Fff**. Процесс переадресации осуществляется через клиента (в этом случае это браузер) сервлета**Ppp**. При выполнении переадресации браузером всегда генерируется запрос типа**GET**.

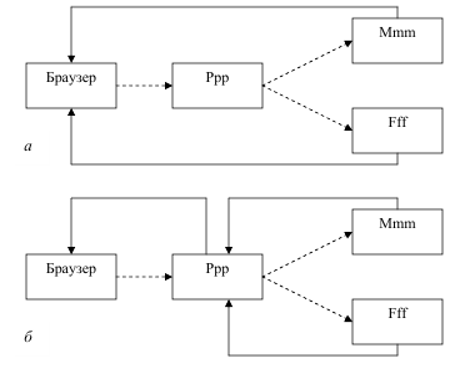
Переадресация выполняется с помощью метода **sendRedirect**объекта типа**HttpServletResponse**. Ресурсом, на который осуществляется переадресация, в общем случае, может быть любой компонент приложения, допускающий запрос типа**GET**.

Доп. Материалы: https://kokoc.com/blog/kak-sdelat-redirekt-podrobnoe-rukovodstvo-po-nastroyke-i-ispolzovaniyu/

1. **Что такое переопределение?**

**Forward**-переопределение, может выполнится, если другой сервлет находится в рамках этого приложения. Связь с первым сервлетом теряется, ответы поступают напрямую браузеру.

Браузер вызывает сервлет Ppp, который в зависимости от значения передаваемого параметра в первом случае переопределяет запрос на сервлеты **Mmm**или**Fff**, при переопределении запроса(RequestDispatcher rd = rq.getRrequestDispatcher("url"); rd.forward(rq,rs);), а во втором – генерирует http-запрос, вызывающий**Mmm**или**Fff**.



Обратите внимание, что при переопределении запроса (a), связь браузера с сервлетом **Ppp**фактически прекращается и данные ответа сервлетов**Mmm**или**Fff**поступают на обработку ему напрямую.

Генерация запросов

+В случае генерации запроса (б), ответы сервлетов **Mmm**и**Fff**поступают и обрабатываются в вызывающем сервлете**Ppp**и только сгенерированный**Ppp**ответ может быть интерпретирован браузером. Внутри сервлета генерируется HTTP-запрос, и теперь этот сервлет будет выступать как клиент для другого сервлета. Ответы поступают назад к первому сервлету и он выдает ответ браузеру. Для генерации http-запросов будем использовать пакет**httpclient**.

1. **Сколько запросов и ответов в обоих случаях?**

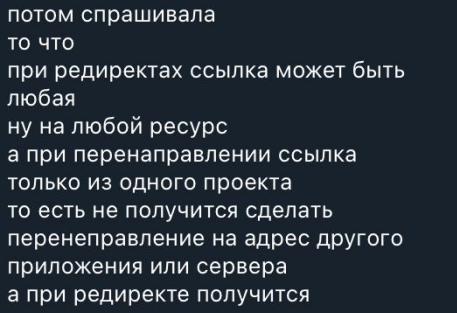
переопределение 1 запрос 1 ответ

переадресация 2 туда и 2 обратно

1. **Кому приходит запросы и ответы в обоих случаях?**

**Переопредление серверу – запросы, клиенту ответы**

**Переадресация – запрос – серверу, ответы браузеру**

**5. Как меняется ссылка?**

**\*При помощи перенаправлений**

**6. Какие бывают сообщения о перенаправлениях?**

Есть несколько типов перенаправлений и делятся на три категории: постоянные, временные и специальные перенаправления.

### [Постоянные перенаправления](https://developer.mozilla.org/ru/docs/Web/HTTP/Redirections#постоянные_перенаправления)

Эти перенаправления призваны длиться вечно. Они подразумевают, что оригинальный URL-адрес больше не должен использоваться, а вместо него должен быть использован новый. Поисковые роботы запускают обновление связанного URL-адреса для ресурса в своих индексах.

| **Код** | **Текст** | **Обработка метода** | **Случаи использования** |
| --- | --- | --- | --- |
| 301 | Moved Permanently | [GET](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/GET) методы неизменны. Другие методы могут быть превращены в [GET](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/GET).[[1]](https://developer.mozilla.org/ru/docs/Web/HTTP/Redirections#attr1) | Реорганизация веб-сайта. |
| 308 | Permanent Redirect | Метод и тело запроса  неизменны. | Реорганизация веб-сайта, с не-GET ссылками/операциями. |

[1] Спецификация не была намерена разрешать изменение метода, но на практике, клиентские приложения делают это. Код 308 был создан чтобы избавиться от неоднозначности в поведении, при использовании не-GET методов.

### [Временные перенаправления](https://developer.mozilla.org/ru/docs/Web/HTTP/Redirections#временные_перенаправления)

Иногда, доступ к запрашиваемому ресурсу не может быть предоставлен из определённого места, но может быть предоставлен из другого. В этом случае, могут быть использованы временные перенаправления. Поисковые роботы не запоминают новую, временную ссылку. Временные перенаправления также используются, когда создаются, обновляются, или удаляются ресурсы, которые представляют временные страницы.

| **Код** | **Текст** | **Обработка метода** | **Случаи использования** |
| --- | --- | --- | --- |
| 302 | Found | [GET](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/GET) методы неизменны. Другие методы могут  быть превращены в [GET](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/GET).[[2]](https://developer.mozilla.org/ru/docs/Web/HTTP/Redirections#attr2) | Веб-страница недоступна по непредвиденным причинам. В этом случае поисковые роботы не будут обновлять свои ссылки. |
| 303 | See Other | [GET](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/GET) методы неизменны. Другие превращены в GET (тело запроса теряется). | Используется для перенаправления после [PUT](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/PUT) или [POST](https://developer.mozilla.org/ru/docs/Web/HTTP/Methods/POST) для предотвращения обновления страницы, что может спровоцировать повторный вызов операции. |
| 307 | Temporary Redirect | Метод и тело запроса  неизменны. | Веб-страница недоступна по непредвиденным причинам. В этом случае поисковые роботы не будут обновлять свои ссылки. Лучше чем код 302 когда не-GET ссылки/операции доступны на сайте. |

[2] Спецификация не была намерена разрешать изменение метода, но на практике, клиентские приложения делают это. Код 307 был создан чтобы избавиться от неоднозначности в поведении, при использовании не-GET методов.

### [Специальные перенаправления](https://developer.mozilla.org/ru/docs/Web/HTTP/Redirections#специальные_перенаправления)

В добавок к обычным перенаправлениям, есть 2 специальные.  Перенаправление с кодом [304](https://developer.mozilla.org/ru/docs/Web/HTTP/Status/304) (Not Modified) перенаправляет страницу к локальной закешированной копии (которая была устаревшей), и перенаправление с кодом [300](https://developer.mozilla.org/ru/docs/Web/HTTP/Status/300) (Multiple Choice) это ручное перенаправление: тело, представленное браузером, как веб-страница, перечисляет возможные перенаправления и пользователь выбирает одно из них.

| **Код** | **Текст** | **Случаи использования** |
| --- | --- | --- |
| 300 | Multiple Choice | Не так много: варианты перечислены на HTML странице. Может быть обслужен со статусом [200](https://developer.mozilla.org/ru/docs/Web/HTTP/Status/200) OK. |
| 304 | Not Modified | Обновление кеша: означает, что значение кеша все ещё актуально и может быть использовано. |

**7.**