

# Page Navigation

Page navigation is triggered by the adapter class. Typically, it is a method call which is triggered by a button click. The `Adapter` class from which you derive your adapter class, offers some methods which make page navigation very simple.

This chapter covers the following topics:

- The First Navigation
  - Preparing the Adapter before Navigating
  - Including the Adapter while Navigating
- 

## The First Navigation

The most simple way of navigating can be seen in the following code example:

```
public void showNextPage()
{
    // check if navigation is possible
    if (... any check is wrong...)
    {
        ...
        this.outputMessage("E","Please first input all fields");
        return;
    }
    // open new page
    this.switchToPage("pageName.html");
}
```

In this method, first there is a check whether navigation is "appropriate" in the current situation. If not, an error message is shown in the status bar. Otherwise, navigation is done by the inherited method `switchToPage(pageName)`.

## Preparing the Adapter before Navigating

In our example, the next screen is - as usual - linked with a specific adapter class. An instance of this adapter class is generated by the session management.

If you want to prepare the adapter of the next screen in a certain way, you proceed as follows.

Before navigating to the next page, you can ask for the adapter which is linked to the next page:

```
public void showNextPage()
{
    // check if navigation is possible
    if (... any check is wrong...)
    {
        ...
        this.outputMessage("E","Please first input all fields");
        return;
    }
    // prepare adapter object which corresponds to next page
```

```
XYZModel m = (XYZModel)this.findAdapter(XYZModel.class);
m.setParam1(...);
m.setParam2(...);
...
...
// open new page
this.switchToPage("pageName.html");
}
```

The method `findAdapter` returns the adapter object which is assigned to the next page. Therefore, you are able to prepare the adapter by setting any information you want to show in the next screen.

## Including the Adapter while Navigating

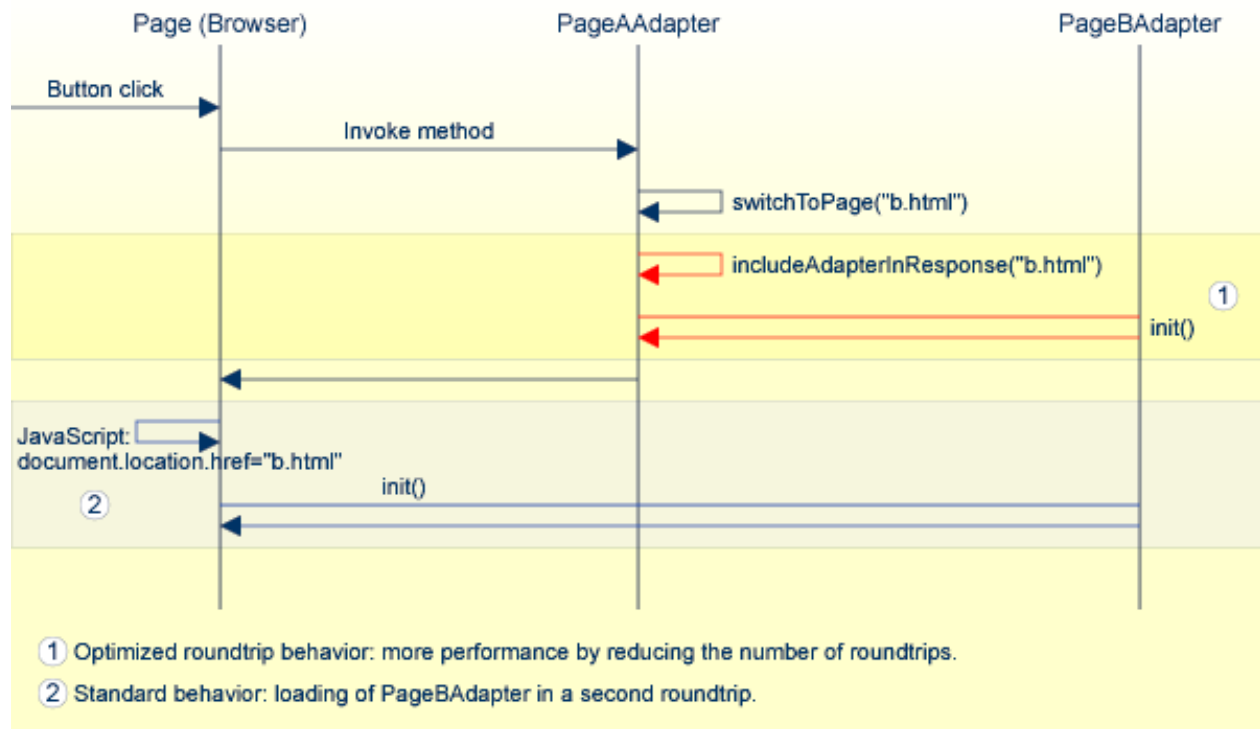
The following example shows how to increase performance for page navigation using the method `includeAdapterInResponse`. It is typically used with:

- `switchToPage()`;
- `openPopup()`;
- `openCISPageInTarget()`

Example:

```
public void showNextPage()
{
    // open new page
    switchToPage("pageName.html");
    // initialize the corresponding Adapter of pageName.html
    includeAdapterInResponse("pageName.html", true);
}
```

The method `includeAdapterInResponse` includes the adapter of the second page (page B) into the response processing of the first page (page A). The adapter is processed in the same way as it is processed when being called by an explicit HTTP request coming from the browser. This is an effective mechanism for reducing the number of roundtrips between the browser and the server (it reduces the number of roundtrips from two to one). This is illustrated by the following diagram:

**Note:**

In a local area network (LAN) environment, the gain in performance will not be significant. However, in a slow wide area network (WAN) environment, the performance will be improved significantly.