

## Base Language - Bug #8346

### Find and fix any use cases with implicit use of working directory when accessing file-system resource

02/26/2024 01:05 PM - Hynek Cihlar

<b>Status:</b> New	<b>Start date:</b>
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b>	<b>% Done:</b> 0%
<b>Category:</b>	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b>	<b>case_num:</b>
<b>billable:</b> No	
<b>vendor_id:</b> GCD	
<b>Description</b>	
<b>Related issues:</b>	
Related to Base Language - Feature #4065: server-side processing of client pl... <b>WIP</b>	

#### History

##### #1 - 02/26/2024 01:18 PM - Hynek Cihlar

- Subject changed from Find and fix any use cases with implicit use of current directory when accessing file-system resource to Find and fix any use cases with implicit use of working directory when accessing file-system resource

File system access through the file system resource was exclusively executed on the client. Since [#4065](#) it can also execute on the server.

With the move a potential issue arises. Use cases involving "implicit" working directory may break when the file system resource is executed on the server.

The goal of this issue is to identify those use cases and come up with a solution. There are several places in the FileSystemOps and FileSystemDaemon where the working directory is assumed. For example see the methods isPathRelativeToCurrentDir, searchPath, getWorkingDirectory, setWorkingDirectory, appendMissedPathEntry.

This issue was identified in #8153-37.

##### #2 - 02/26/2024 01:19 PM - Hynek Cihlar

- Related to Feature #4065: server-side processing of client platform dependencies added