

Handling exit status

Most LMX API functions return status information LMX_STATUS that indicates the success or failure of the requested operation. You should review this information to be able to more easily and accurately identify the problem.

The following code is an example of an error handler that writes errors to stderr. Additionally, it is used in all code snippets provided in subsequent sections to make them more concise. Please make sure this function is defined if you want to compile the code as it stands.

```
LMX_HANDLE h;

void exit_on_error(LMX_STATUS s)
{
    if (s != LMX_SUCCESS)
    {
        if (h != NULL)
        {
            fprintf(stderr, "%s\n", LMX_GetErrorMessage(h));
            LMX_Free(h); // Release memory allocated by LM-X
        }
        else
        {
            fprintf(stderr, "%s\n", LMX_GetErrorMessageSimple(s));
        }
        fflush(stderr);
        exit(1);
    }
}
```

On failure, when a function returns the status code other than LMX_SUCCESS, the above example will display the corresponding error and its context, as described in [LMX_GetErrorMessage](#). If the handler is null, it will only display the error name (see [LMX_GetErrorMessageSimple](#)).