

```
#else
int main(int argc, char ** argv)
#endif
{
    int weight, queue, rc;
    char cmd_buf[MAX_LINE_LEN];
    ag_uint32_t parameters[4];
    ag_uint8_t param_nums[]={LIMIT1_NUM, LIMIT2_NUM, LIMIT3_NUM, EXPENSE_NUM};
    char * filename;
    FILE * cfg_fp;
    ag_chipset_chip_error_t chipError;

    /* get and check arguments */
    if (argc != 2) {
        fprintf(stderr, "Usage: DiffServStart <cfg file name>\n");
        exit(-1);
    }
    filename = argv[1];
    if ((cfg_fp = fopen(filename, "r")) == NULL) {
        fprintf(stderr, "Error: can not open config file %s\n", filename);
        exit(-1);
    }

    /* initialize chipset */
    rc = ag_chipset_init(0, &chipsetHdl);
    if (rc != AG_CHIPSET_ST_SUCCESS) {
        fprintf(stderr, "Error: ag_chipset_init failed\n");
        fclose(cfg_fp);
        exit(-1);
    }

    /* configure chipset */
    rc = ag_chipset_config(chipsetHdl, (ag_read_fn_t) cfg_read_func,
                          (void *) cfg_fp, NULL, NULL, &chipError);
    if (rc != AG_CHIPSET_ST_SUCCESS) {
        fprintf(stderr, "Error: ag_chipset_config failed\n");
        fclose(cfg_fp);
        exit(-1);
    }
    /* close config file */
    fclose(cfg_fp);

    /* get object-level APP550 handle */
    rc = ag_chipset_hdl_get(chipsetHdl, (char*) "APP550", 0, &app550Hdl);
}
```

```

else if (is_nav3) is_js = 1.1;
else if (is_opera5up) is_js = 1.3;
else if (is_opera) is_js = 1.1;
else if ((is_nav4 && (is_minor <= 4.05)) || is_ie4) is_js = 1.2;
else if ((is_nav4 && (is_minor > 4.05)) || is_ie5) is_js = 1.3;
else if (is_hotjava3up) is_js = 1.4;
else if (is_nav6 || is_gecko) is_js = 1.5;
// NOTE: In the future, update this code when newer versions of JS
// are released. For now, we try to provide some upward compatibility
// so that future versions of Nav and IE will show they are at
// *least* JS 1.x capable. Always check for JS version compatibility
// with > or >=.
else if (is_nav6up) is_js = 1.5;
// NOTE: ie5up on mac is 1.4
else if (is_ie5up) is_js = 1.3

// HACK: no idea for other browsers; always check for JS version with > or >=
else is_js = 0.0;

// *** PLATFORM ***
var is_win = ( (agt.indexOf("win")!=-1) || (agt.indexOf("16bit")!=-1) );
// NOTE: On Opera 3.0, the userAgent string includes "Windows 95/NT4" on all
// Win32, so you can't distinguish between Win95 and WinNT.
var is_win95 = ((agt.indexOf("win95")!=-1) || (agt.indexOf("Windows 95")!=-1));

// is this a 16 bit compiled version?
var is_win16 = ((agt.indexOf("win16")!=-1) ||
               (agt.indexOf("16bit")!=-1) || (agt.indexOf("Windows 3.1")!=-1) ||
               (agt.indexOf("Windows 16-bit")!=-1) );

var is_win31 = ((agt.indexOf("Windows 3.1")!=-1) || (agt.indexOf("win16")!=-1)
||
               (agt.indexOf("Windows 16-bit")!=-1));

var is_winme = ((agt.indexOf("win 9x 4.90")!=-1));
var is_win2k = ((agt.indexOf("windows nt 5.0")!=-1));

// NOTE: Reliable detection of Win98 may not be possible. It appears that:
// - On Nav 4.x and before you'll get plain "Windows" in userAgent.
// - On Mercury client, the 32-bit version will return "Win98", but
// the 16-bit version running on Win98 will still return "Win95".
var is_win98 = ((agt.indexOf("win98")!=-1) || (agt.indexOf("Windows 98")!=-1));
var is_winnt = ((agt.indexOf("winnt")!=-1) || (agt.indexOf("windows nt")!=-1));
var is_win32 = (is_win95 || is_winnt || is_win98 ||
               ((is_major >= 4) && (navigator.platform == "Win32")) ||
               (agt.indexOf("win32")!=-1) || (agt.indexOf("32bit")!=-1));

var is_os2 = ((agt.indexOf("os/2")!=-1) ||

```