

```

/*
+-----+
+-----+
|      Copyright (c) 1996-1999 Vail Systems, Inc.  All Rights
Reserved      |
+-----+
+-----+
$Id: mc_vm.c,v 1.139 1999/12/31 16:30:12 alex Exp $
+-----+
+-----+
*/
/
*****
*****

Module : mc_vm.c

This module holds Webley application code.

HISTORICAL INFORMATION:

10-04-96  Alex Kurganov:Vail  Created

*****
*****/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <signal.h>
#include <sys/types.h>
#include <sys/timeb.h>
#include <time.h>
#include <dirent.h>
#include <sys/wait.h>
#include <sys/stat.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <fcntl.h>
#include <memory.h>
#include <setjmp.h>
#include <unistd.h>
#include <math.h>
#include <errno.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#include <varargs.h>
#include <sys/time.h>

```

```

#include "vlib_scsa.h"
#include "notif_scsa.h"
#include "app_api.h"
#include "mcall.h"
#include "mc_conf.h"
#include "mc_br_result.h"
#include "webley_vocab.h"
#include "app_lnhttps.h"          /* internal TTS definitions */

#undef  DEBUG
#define DEBUG(x)  x

#define MAX_LINE_LEN  101

#define SOCKET          int
#define REM_SERV_VAR_LENGTH  512
#define REM_SERV_NO_MAX    16
#define SECONDS          20
#define MX_HOST_NAME_SIZE  32
#define MAXBUFFER        2*4096
#define TEMPBUF          2*4096
#define APACHE_PORT      80
#define UNISERV_PORT      6666
#define PORT_IN_USE       UNISERV_PORT

/* Confidence measures for Nuance 6.11 */
#define GOOD_BYE_CONFIDENCE  400
#define DELETE_CONFIDENCE    400
#define NAME_CONFIDENCE      450

SOCKET make_connection();
char *GetConfigVar(char*, char*);

/* This is the prefix for the */
/* temporary recording file */

#define REPEAT_PLAYMENU      2
#define MENU_POSITIONS      5

#define F_FORK_CHILD         1
#define F_NO_FORK            0

static char vr_buf[144];
static char vr_tok[144];
static int msgArc = 0;
static int dSeen = 0;
static int globalVRmode = 0;

int SessionJobNumber = -1;
static int confNum = -1;
static char getMname_msg[128];
static int scodeValid = 0;

```

```

static int classOfService = 0;
int playFullFlag = 0;
int playFullFaxFlag = 0;
char *playList[MAX_MULTPLAY_LIST +1];
int playListCnt = 0;
int dialCnt = 0;

extern int dFlag;
extern int trunk_num;
extern char app_path_name[];
extern char app_name[];
extern char app_home[];
extern char msg_home[];
extern char msg_host[];
extern char *convertPhone();
extern char *vr_convert();
extern int vrAlloc();
extern int vrDealloc();
extern char *    getFoundSubscriber();

extern char    getCreditWarn();
extern double  getPrepaidAmount();
extern double  getWarnThreshold();
extern double  getCreditLimit();
extern double  getStopThreshold();
extern char    getCreditWarn();
extern char    getCreditStop();
extern char    getIntlDial();
extern int     closeDBHandle();
extern DESTINATION_S *callContact();
extern char *  getRandomResponse();
extern DV_TPT rtptarr[MAX_TPTARR];
extern int curr_rtpt;
MESSAGE_S * recordWebleyMsg();
LIST_S * getListByName();
LIST_S * getList();
int phoneSetup(MAILBOX_S *m, int type);
int control_vr_timeout(struct VR_RES *vr_tokens);
void trimMessage(char *fname, int sampRate, int *msgLength);
extern int isOutbound();

extern char *  getResellerAccount();

#ifdef PBELL_PERSONAL
    char mpack_module[] = "pacbellMpack.pl";
    char mpack_module_trans[] = "pacbellMpackTrans.pl";
#else
    char mpack_module[] = "webleyMpack.pl";
    char mpack_module_trans[] = "webleyMpackTrans.pl";
#endif
/

```

```

* -----
----- */
int setServiceClass(int servClass)
{
    classOfService = servClass;
    return classOfService;
}
/
* -----
----- */
int getClassOfService()
{
    return classOfService;
}
/
* -----
----- */
int getScode()
{
    /* return scodeValid; */
    return 1;
}
/
* -----
----- */
int callingFromBase(MAILBOX_S *m)
{
    DESTINATION_S *cd;
    if ( m ) {
        cd = checkDestination(m, getCallerPhone(),
MC_PERSONAL_GROUP);
        if (!cd)
            return R_FALSE;
        switch(cd->dType) {
            case MC_HOME_TYPE :
            case MC_OFFICE_TYPE :
            case MC_MOBILE_TYPE :
            case MC_CELL_TYPE :
                return R_TRUE;
        }
    }
    return R_FALSE;
}
/
* -----
----- */
int setMsgArc(arc)
{
    msgArc = arc;
    return R_TRUE;
}
/

```

```

* -----
----- */
int getMsgArc()
{
    return msgArc;
}
/
* -----
----- */
/* This routine is used to disable DTMF interrupts for conference
on the fly
* on an inbound side when a subscriber calling another
subscriber
*/
int talkingToSubscriber(char *mode)
{
    static int talkSubFlag = 0;

    if (!strcmp(mode, "set"))
        talkSubFlag = 1;
    else if (!strcmp(mode, "clear"))
        talkSubFlag = 0;
    return talkSubFlag;
}
/
* -----
----- */
int isName(char *name)
{
    register int i, len;
    if (!name || !*name )
        return 0;
    len = strlen(name);
    for (i = 0; i < len; i++) {
        if (!isdigit(name[i]) )
            /* it looks like name, not a speed-dial code */
            return 1;
    }
    return(0);
}
/
* -----
----- */
char * getAssistantNoReplyAddress()
{
#ifdef PBELL_PERSONAL
    return "no_reply@pacbell-pa.com";
#else
    if (!strcmp((char *)getResellerAccount(), "5800" ))
        return "no_reply@opusassistant.com";
    else
        return "no_reply@webley.com";
#endif
}

```

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.