

# CCSS NEWS AND UPDATES

Pro-Active Systems Management Solutions for the IBM System i

AUGUST 2006

ISSUE 18

## COMPANY NEWS



### CCSS Expands And Moves North American Office And Relocates Product Manager From UK

CCSS, developers of the QSystem Management Suite of monitoring and messaging software for the IBM System i (formerly known as the iSeries and AS/400), has expanded and moved its North American office in Raleigh, North Carolina, and relocated Product Manager Paul Ratchford from the U.K. to the U.S.A.

"While the Internet and the telephone certainly have shrunk the distance between Europe and the United States, nothing beats face-to-face meetings with customers. These will be easier to arrange now that I am in Raleigh," says Product Manager Paul Ratchford. "A major benefit of such meetings is watching our product at work in different environments. This helps us continuously improve our solutions. Another benefit is being closer to Rochester and the IBM team." [Click here for full details](#)

## CUSTOMER REMINDER

Keep your iSeries system up to date with the latest critical support updates, features and security updates. Upgrade all CCSS products to the latest releases available through the secure technical support section of the CCSS website. For any technical assistance please contact [uk.support@ccsseurope.co.uk](mailto:uk.support@ccsseurope.co.uk)

LAST GA DATE : 10.08.06  
<http://www.ccsstld.com/support/downloadsoftware.jsp>

PRODUCT	RELEASE	MODIFICATION LEVEL
QSystem Monitor	11	140
QMessage Monitor	6	63
QRemote Control	2	53

## TECHNICAL TIP



### Starting Fresh By Kurt Thomas

For every iSeries operator, there comes a time when he or she needs to say goodbye – goodbye to the hustle and bustle of the many, many active threads on his iSeries and the stoic patience of the hundreds or thousands of jobs waiting to get their spooled files out; a time when he has to let go of scores of invisible temporary objects whose presence he can but sense; a time, in short, when he powers down the iSeries. Then only the whir of the air-conditioning fills the computer room.

But not for long! The system is powering up again! Server processes rise like phoenix from the ashes, the work control table and many other invisible helpers are recreated, and a flurry of activity once again fills the system, reminiscent of nothing more than of that first day in spring when you decide to clean the house (which you then immediately postpone for two weeks).

Amidst all this electronic rejuvenation, MSM and MM are born - that is, if you have taken care to modify your start-up program. In this month's installment of "All Things Spring", we'll show you how to do just that.

For starters, use `DSPSYSVAL SYSVAL(QSTRUPPGM)` to get the library and object that contain the startup program. Then use `DSPPGM` to determine the source library, file, member for the program. In 99% of all cases, the program is a CL program and the source code for it is on the system. We'll assume you use TCP/IP, as that's what the majority of our customers use. Create a copy of the source, and add the following lines to the copy:

```
/* *** Start QSystem Monitor *** */
ADDLIB LIB(MSM) /* Add MSM lib to libl */
MONMSG MSGID(CPF0000) /* Monitor for error msgs */
MONSTRTCP STRSBS(*SBSONLY) /* Start MSM TCP server+ */
/* + and subsystem only */

MONMSG MSGID(CPF0000)
MONBCK /* Start MSM monitoring */
MONMSG MSGID(CPF0000)

RMVLIB LIB(MSM) /* Take MSM out of libl again */
MONMSG MSGID(CPF0000)

/* *** Start QMessage Monitor *** */
ADDLIB LIB(MM) /* Add MM lib to libl */
MONMSG MSGID(CPF0000) /* */
```

```

MMRESET SYST(*ALL) PC(*YES)          /* Resets internal status+ */
                                        /* for all systems and msg */
                                        /* queues monitored by MM; */
                                        /* useful after abnormal */
                                        /* terminations of MM      */
MONMSG      MSGID(CPF0000)
MMSTRTCP          /* Start the MM TCP server */
MONMSG      MSGID(CPF0000)
MMSTRMTR GRP(*DFT) STRQ(*YES)        /* Start MM monitoring for+ */
                                        /* all msg queues           */
MONMSG      MSGID(CPF0000)
RMVLIBLE LIB(MM)
MONMSG      MSGID(CPF0000)

```

If you use the startup program to start TCP/IP, make sure to put in the above lines after TCP/IP is started. It may also be useful to add in a DLYJOB to make sure TCP/IP is fully started.

Note the near-obsessive use of MONMSG in the code. If you are new to modifying the startup program, this will seem a bit odd, but it's the way to go, because you *don't* want the startup program to stop when an error occurs.

After modifications to the source, you can compile the program and change system value QSTRUPPGM to point at the modified program.

After the next IPL, you can check the logs for errors by running: WRKSPLF SELECT(QPGMR \*ALL \*ALL QSTRUPJD).

If you would like any additional technical assistance please contact us at [support@ccsseurope.co.uk](mailto:support@ccsseurope.co.uk) and we will be happy to assist you.



As an existing customer of CCSS, you will continue to receive our regular email newsletter. If you would prefer not to receive future copies, please send us an email to [news@ccssltd.com](mailto:news@ccssltd.com) with the word **UNSUBSCRIBE** in the subject heading.

<http://www.ccssltd.com>

© 2006. All Rights Reserved.