S/390 common I/O-Layer - command line parameters, procfs and debugfs entries

Command line parameters

* ccw_timeout_log

  Enable logging of debug information in case of ccw device timeouts.

* cio_ignore = {all} |
  {<device> | <range of devices>} |
  {!<device> | !<range of devices>}

  The given devices will be ignored by the common I/O-layer; no detection
  and device sensing will be done on any of those devices. The subchannel to
  which the device in question is attached will be treated as if no device was
  attached.

  An ignored device can be un-ignored later; see the "/proc entries"-section for
  details.

  The devices must be given either as bus ids (0.x.abcd) or as hexadecimal
  device numbers (0xabcd or abcd, for 2.4 backward compatibility). If you
  give a device number 0xabcd, it will be interpreted as 0.0.abcd.

  You can use the 'all' keyword to ignore all devices.
  The '!' operator will cause the I/O-layer to _not_ ignore a device.
  The command line is parsed from left to right.

  For example,
        cio_ignore=0.0.0023-0.0.0042,0.0.4711
  will ignore all devices ranging from 0.0.0023 to 0.0.0042 and the device
  0.0.4711, if detected.
  As another example,
        cio_ignore=all,!0.0.4711,!0.0.fd00-0.0.fd02
  will ignore all devices but 0.0.4711, 0.0.fd00, 0.0.fd01, 0.0.fd02.

  By default, no devices are ignored.

/proc entries

* /proc/cio_ignore

  Lists the ranges of devices (by bus id) which are ignored by common I/O.

  You can un-ignore certain or all devices by piping to /proc/cio_ignore.
  "free all" will un-ignore all ignored devices,
  "free <device range>, <device range>, ..." will un-ignore the specified
  devices.

  For example, if devices 0.0.0023 to 0.0.0042 and 0.0.4711 are ignored,
  - echo free 0.0.0030-0.0.0032 > /proc/cio_ignore
    will un-ignore devices 0.0.0030 to 0.0.0032 and will leave devices 0.0.0023

  to 0.0.002f, 0.0.0033 to 0.0.0042 and 0.0.4711 ignored;
- echo free 0.0.0041 > /proc/cio_ignore will furthermore un-ignore device
  0.0.0041;
- echo free all > /proc/cio_ignore will un-ignore all remaining ignored
  devices.

When a device is un-ignored, device recognition and sensing is performed and the device driver will be notified if possible, so the device will become available to the system. Note that un-ignoring is performed asynchronously.

You can also add ranges of devices to be ignored by piping to /proc/cio_ignore; "add <device range>, <device range>, ..." will ignore the specified devices.

Note: While already known devices can be added to the list of devices to be ignored, there will be no effect on then. However, if such a device disappears and then reappears, it will then be ignored. To make known devices go away, you need the "purge" command (see below).

For example,
"echo add 0.0.a000-0.0.accc, 0.0.af00-0.0.afff > /proc/cio_ignore"
will add 0.0.a000-0.0.accc and 0.0.af00-0.0.afff to the list of ignored devices.

You can remove already known but now ignored devices via
"echo purge > /proc/cio_ignore"
All devices ignored but still registered and not online (= not in use) will be deregistered and thus removed from the system.

The devices can be specified either by bus id (0.x.abcd) or, for 2.4 backward compatibility, by the device number in hexadecimal (0xabcd or abcd). Device numbers given as 0xabcd will be interpreted as 0.0.abcd.

* /proc/cio_settle

A write request to this file is blocked until all queued cio actions are handled. This will allow userspace to wait for pending work affecting device availability after changing cio_ignore or the hardware configuration.

* For some of the information present in the /proc filesystem in 2.4 (namely, /proc/subchannels and /proc/chpids), see driver-model.txt.
Information formerly in /proc/irq_count is now in /proc/interrupts.

debugfs entries
---

* /sys/kernel/debug/s390dbf/cio_*/ (S/390 debug feature)

Some views generated by the debug feature to hold various debug outputs.

- /sys/kernel/debug/s390dbf/cio_crw/sprintf
  Messages from the processing of pending channel report words (machine check handling).

- /sys/kernel/debug/s390dbf/cio_msg/sprintf

  Various debug messages from the common I/O-layer.

- /sys/kernel/debug/s390dbf/cio_trace/hex_ascii
  Logs the calling of functions in the common I/O-layer and, if applicable,
  which subchannel they were called for, as well as dumps of some data
  structures (like irb in an error case).

The level of logging can be changed to be more or less verbose by piping to
/sys/kernel/debug/s390dbf/cio_*/level a number between 0 and 6; see the
documentation on the S/390 debug feature (Documentation/s390/s390dbf.txt)
for details.