

THE HEXADECIMAL KID AND HIS FAITHFUL DOG ASCII

=====

[A Fantasy in Sixteen Bits]

Copyright (C) 1977, Richard Forsyth.

Bit 5 (A Bit More)

[By a deception, Dr Null has gained control over the Hexadecimal Kid's dog Ascii. He has used Ascii's sharp teeth to cut their communication link with the outside world, and now Hex and his friends are trapped. He is insisting that they aid him in his efforts to destroy the Database (and hence civilization). With Ascii's bionic teeth and laser-beam eyes to back him up, they may find it hard to refuse.]

"All right," said Hex after a long pause. "What do you want us to do?"

"I am glad you have seen reason at last," responded Dr Null with satisfaction, "though it does seem to take rather a long time for your computerized minds to accept the obvious. Your task is straightforward. I have been so impressed by your dog Ascii that I have decided to alter my plans slightly to make use of his special abilities. I now feel that it would be a waste of time to degrade the Database slowly by annulling data in transmission. It is preferable to strike at the very heart of the Network and annihilate it at a stroke. It seems to me that Ascii is admirably suited for this purpose." He patted the dog affectionately as he spoke. "I propose to transform him into an informatic doomsday missile, launched at the Central Processor itself, carrying in his shaggy warhead enough false information to disable the nerve centre of the Network and bring its activities grinding to a standstill. In the confusion the Database will be defenceless, and we can erase its contents at our leisure."

Hex could have sworn that Ascii's eyes lit up at the prospect of this starring role in such an epoch-making event.

"All I need from you is a full description of the Executive's data-vetting procedures so that Ascii can pass through the inner security ring, and some help in adapting him for his new function. I have written a program that will put the System into a gigotic loop. Ascii can deliver it."

Hex shuddered. It was worse than he had feared. If the mad doctor's fiendish scheme succeeded, the Network would be paralyzed. Hex reckoned that there was enough distributed intelligence in the System for it to survive such an assault, but only at the cost of a massive loss of data and the permanent inactivation of many millions of androids and robots. Even the Great Crash would appear trivial beside a disaster on such a scale.

The really alarming part was the idea of putting the System into a gigotic loop. The mathematics of gigotic processes had been worked out by Professor Synapse himself, shortly before his mysterious death. It was hardly surprising that his son should know of them.

Professor Synapse had proved that any computing process -- however rigorously tested -- could under certain conditions become gigotic, entering a kind of positive feedback cycle which repeatedly magnified an originally tiny error. Although the chances of any given

process going gigotic were negligible, in the Network millions of jobs were processed every second. Many researchers had worked on the problem of how to prevent gigotic processes arising and how to check them if they occurred, but there was no generally effective protection. Once a process became gigotic, there was no way to restore it to normal operation; furthermore, any processes with which it interacted also became gigotic by the phenomenon known as gigotic induction. The idea of starting one deliberately (and in the centre of the Network) was positively diabolical.

The peculiar vulnerability of the Database to gigotic instability arose from two facts: firstly, the Network was universal and all processes were potentially linked, so it would spread like a plague if it ever took hold; and secondly, the System had to be live and running at all times. If a robot or an android became gigotic it could, in the last resort, simply be switched off; but switching off the Central Processor was out of the question. In any case such a drastic step would, of course, have to be authorized by the Executive, which was by definition out of action -- the classic cache 22.

Hex forced himself to say: "Right, let's get started." His one idea was to play for time.

"Excellent!" said Dr Null.

Simula just glared at him. Fetch and Execute looked puzzled too. Hex wanted to broadcast to them on the UHF channel, saying "don't worry", but he dared not since Dr Null was evidently capable of tapping their private frequency. He just hoped Simula would deduce that he was bluffing.

Dr Null started to give orders. "First of all, the Kid can explain to me the workings of the inner kernel of the Executive."

Hex took a piece of paper and began to draw.

UNOS (Unified Network Operating System)		
FATHER (File And Transaction Handling Executive Routine)	SON (System Organizing Nucleus)	GHOST (General High-level Optimal Scheduling Technique)

"The Executive (UNOS) is divided into three main subsystems: FATHER controls all Database access, as well as input and output; GHOST is responsible for queuing and resource allocation; and SON supervises all other processes in the system. The protection mechanisms are in the SON."

"I know all this," cut in Dr Null impatiently. "I want details."

"Well if you come to my workshop," replied Hex, "I can show you a copy of the System Manager's Handbook, complete with flowcharts, compilation listings, cross-reference tables, bug-fix reports -- the lot."

"Lead the way," said Dr Null, practically drooling at the prospect.

"You can't let him see those," cried Simula, appalled at Hex's abject capitulation.

"Quiet!" commanded Dr Null. "The rest of you will remain here, guarded by Ascii. One false move, and he will be on you before you can type control-G. So be warned."

Hex left the room, followed by Dr Null who kept his ultra-violet PROM-eraser pointing straight at Hex's head.

- Is Hex a traitor to the Network?
- Has Ascii contracted the dreaded RABIES (Random Access Binary Information Extermination Syndrome)?
- Wait for our next mind-boggling episode.