1 General Comments on "HPCC and Java". DBC, 03/08/96

- Section 6 about compilation issues is important. If Rice can make a plausible case that Java compilers will eventually attain the efficiency close to C compilers (as Sun promise us) the project looks plausible. If that doesn't happen the whole idea of parallel HP Java looks lame.
- Texas (sect 1.6) and Indiana (sects 1.6, 3.1) both downplay the importance of data parallelism (to be fair, they only say that adding language extensions to support data parallelism is a bad idea).

  This seems counterproductive. I would be more comfortable if we were embracing data parallelism. Once we've convinced ourselves that Java is fast, there's no reason not to do number-crunching in it.
- Some vision (if we have one) for applications of HP Java probably ought to go before "Suitable Models of Parallel Computing".
- In "Suitable Models of..." the most obvious suitable model is the conservative one of straightforward message passing between processes implemented as applets (say), with class libraries for channels (or something). This can be made more interesting by emphasizing that the processes are dynamically loaded onto Java servers, or cooperating Web servers (or something) in a heterogeonous network, with run-time creation of extra remote applets as they are needed (or something).

  This is not to downplay the parallel object ideas from Texas, which may well be more interesting in the long term. But it isn't clear to me that all of us can operate in that way.

  Maryland contribution to 7.4.1 is also relevant to this section?
- Section 3 absorbed into section 4?
- Merge sections 5 and 6?
- We definitely need something in section 7.3, "Task parallelism and Java". Fortran-M-like channels would be nice.
- The Maryland contributions: 7.4.1 about itinerant processes seems to be defining or extending a "Suitable Model of Parallel Computing". Should move it into the appropriate earlier section?

  The Maryland contributions 7.4.2 and 7.4.3 seem to be closely related to what ought to be in 7.3.
- The division between "low-level" and "high-level" runtime issues isn't particularly clear. At the moment it isn't obvious that the contents of section 7 are at a higher level than section 8.