Questions for the Round Table

The multi-core is an enabling technology for space but some topics have to be investigated:

1. What is the most suitable application class for NGMP?
   - In Guidance, Navigation and Control systems for critical Entry, Descent and Landing or Formation flying phases?
   - As P/L Computer core processor for Science Missions?
   - Would it make sense to combine platform and payload processing on the same chip?
   - Others ?

2. Selection of the reference benchmarks (as taken out from applications) from the presentations
   - It is important to mimic the target workload of an application, thus enabling to measure the suitability of an architecture for a given application.
   - Benchmarks are also useful for performing relative comparison of architectures: for this there must be agreement between all the players on what to consider.
   - Different benchmarks (AG, BSC, Astrium, …) have been presented, are these suitable for your needs ? Are different/additional features desired?

ESA UNCLASSIFIED – For Official Use

Questions for the Round Table

3. Selection of operating system(s)

- *a) AMP*
  - **A copy of RTEMS on each core**
    - Not suitable for data intensive payload
    - Can run legacy code
    - Task allocation Requires a consistent effort from the programmer
  - Ostrales ?
- *b) SMP*
  - Linux … but complex qualification
  - VxWorks: proprietary
- *c) IMA/Hypervisor*
  - Enables spatial isolation among the partitions, but still requires OSes
  - **XstratuM**
  - PikeOS: proprietary
- *d) Ad-hoc, manual task management*
  - Application code on bare hardware
  - Flexible, always possible but … complex, error prone

ESA UNCLASSIFIED – For Official Use

Questions for the Round Table

4. SW deployment and design model

Are traditional SW design methods applicable directly in case of multi-core?

- How much the SW deployment on multiple cores has to be transparent during SW design?
- Can SAVOIR SW Reference Architecture help? How?
- In case of heterogeneous applications centralised on a multi-core processor, how does the role of the avionics engineer evolve?

5. Selection of the programming language(s)

Standard programming languages, such as C/C++ are not suitable for parallel programming

- Many extensions/new languages exist to tackle the issue: OpenMP, OpenCL, Cilk, MPI. What is the most suitable?
- ADA natively supports multi-core: is its tasking mode also suitable for payload?
- Which degree of control on the software/hardware coupling is needed?

  Tighter control means more control on performance / task distribution, etc. but it puts more burden on the programmer.

ESA UNCLASSIFIED – For Official Use

Questions for the Round Table

6. Open source development?

- Commercial SW tools and OS have good support, but allow limited control on product evolution (and licenses are expensive).
- Open source SW allows more control, but we are “on our own” regarding support and evolutions.

7. Predictability vs performance

*Worst case, to fall under a normal, predictable system, we would have to switch-off all but one core of the system (*not to discuss issues on caches, etc. …)

- To what degree is predictability needed?
- Usually, only small portions of platform applications have HRT requirements: would degrading system performance only for those portions be acceptable?

8. Simulator(s)

The processor emulator is the core of SW Validation Facilities and an important building block of Operational Simulators.

- How can SW execution on multi-core simulated with accuracy and sufficient performance w.r.t. SVF or Operational Sim needs?
- Is performance “faster-than-real-time” a feasible objective?

ESA UNCLASSIFIED – For Official Use

Questions for the Round Table

9. Can we reuse legacy SW on multi-core

   It is desirable to reuse existing SW when available, but…

   - Can this be reused on multi-core with minor intervention?
   - Legacy SW development tool should be compatible with new multi-core architecture (e.g. compilation tool chain, RTOS).
   - Is automatic parallelisation of existing sequential SW interesting?

10. Can the intrinsic redundancy of multi-cores be used to increase the processor's reliability and availability?

    - As a P/L Computer processor core outages of the multi-core SoC during the mission lifetime are possible but for critical phases if used as central core of a GNC systems only controlled degraded operative modes in case of failure are admitted.
    - Implementation of TSP techniques can improve the reliability/availability of multi-core architecture.

ESA UNCLASSIFIED – For Official Use