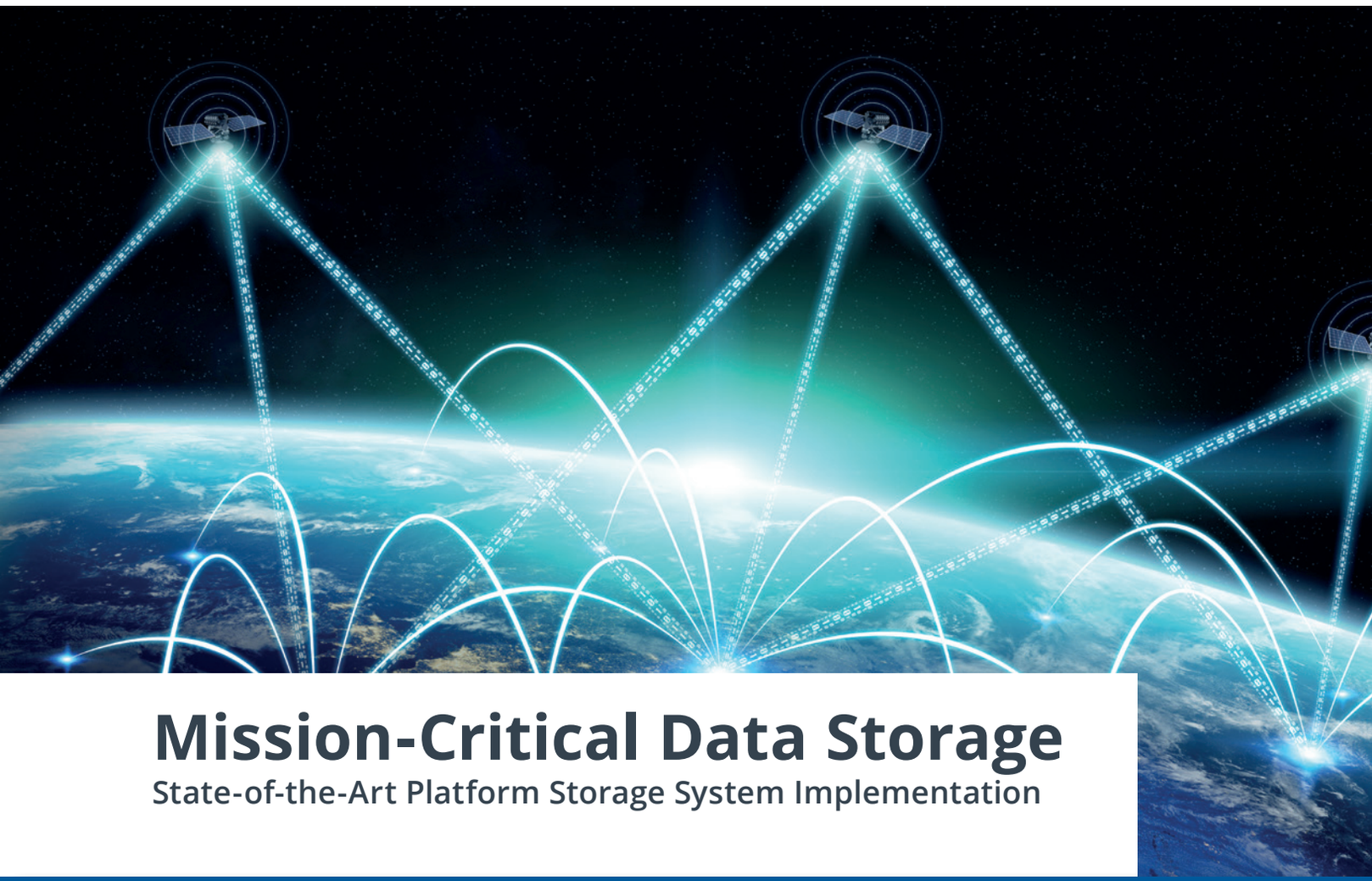

Mission-Critical Data Storage

State-of-the-Art Platform Storage System Implementation

PikeOS RTOS & Hypervisor

Space Systems ready for Future Missions

One of the major objectives during space missions consists of data acquisition. On the technical side, this involves storage and transfer to the ground station. In the past, packet-oriented data structures have widely been used. As consequence, recorded data had been sent to the ground station in a sequential and inflexible way.

Future missions with a huge demand on data transfer may significantly benefit from prioritization based on meta data information. This is where file-based systems come into play, as those also provide better visibility on the data stored by using folders to group the information. The meta data assigned to a file or folder may involve the date of creation and modification, data type and access policies.

Also, most recent space missions are using platform storage not only for scientific data but also to hold mission-critical items like telecommand timelines, application software (ASW) images as well as additional system configuration files.

Figure 1 shows the proposal of a state-of-the-art approach for implementing a platform storage system. The core of the system is the file management service (FMS), which provides its services to several clients, such as scientific applications or the file transfer service (FTS). The latter might exemplarily base on the file delivery protocol (CFDP) as specified by the consultative committee for space data systems (CCSDS) and is not scope of the FMS itself. In the present use case, the file management service is linked against the PikeOS certifiable file system (CFS).

The certifiable file system (CFS) is a PikeOS component that provides a failsafe file system with more functionality than the PikeOS native file system. In addition to the basic file operations implemented by the internal PikeOS file providers (open, close, read, write, map, ioctl, lseek, stat), the CFS can also handle directories and file manipulations (create, delete, rename, truncate, chmod). It is still a simplified file system compared to standard Linux file systems. The CFS supports metadata for files and folders.

The CFS can be mounted and unmounted at runtime. CFS is designed to run on block devices with a configurable block size. The block access service (BAS) is implemented in the PikeOS kernel space. Permanent as well as volatile memory block drivers are available. Memory is usually protected by additional Cyclic Redundancy Check (CRC) bits in order to correct Single Event Upsets (SEU).

This use case just provides a basic overview about the implementation of the data storage system (DSS) for space missions. In reality there are additional topics to be considered. For example, a satellite does not only require storage for the platform itself (PFDS), but also for the payload systems (PLDS).

Please feel free to contact SYSGO's engineers for more information on how to effectively use PikeOS within a complex distributed system or how to support SafeGuard Memory (SGM).

PIKEOS SOFTWARE ARCHITECTURE

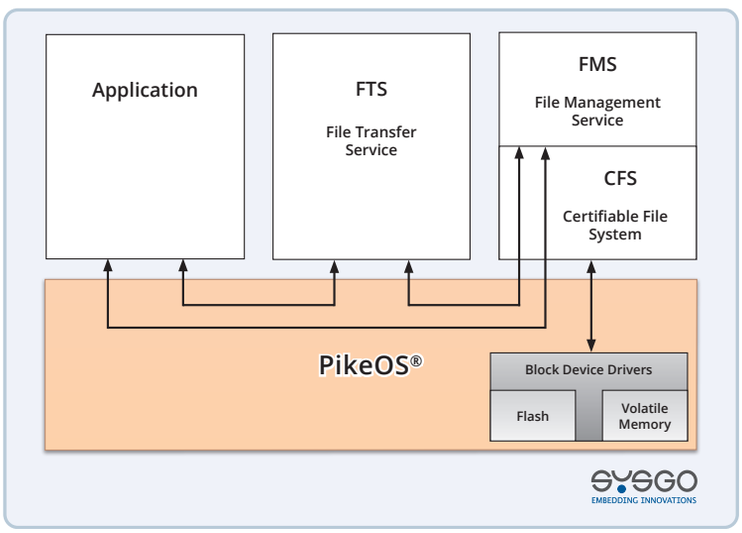

Figure 1

Founded in 1991, SYSGO became a trusted advisor for Embedded Operating Systems and is the European leader in hypervisor-based OS technology offering worldwide product life cycle support. We are well positioned to meet customer needs in all industries and offer tailor-made solutions with highest expectations in Safety & Security. More information at www.sysgo.com/space

SYSGO Headquarters
Phone +49 6136 9948 500
sales-de@sysgo.com

SYSGO France
Phone +33 1 30 09 12 70
sales-fr@sysgo.com

SYSGO Czech Republic
Phone +420 222 138 111
sales-cz@sysgo.com

www.sysgo.com