ARKA

SOFTLINK® SOFTWARE DEVELOPMENT KIT

Create applications for signal, network, and data processing

Reduce time for development and testing

Benefit from increased software maturity

SOFTLINK SwD LIBRARY FOR BUILDING APPS

Our expanding library of Software Devices (SwDs) provides the functional building blocks that are combined and ordered to build Apps. Using Python scripts or JSON definitions, Apps can be built for the Cloud, hosted servers, or hybrid deployments.

GENERAL PURPOSE SwDS

- Network Protocols
- Data Buffering and Queueing
- Processing Threads
- Time-Data Correlation
- Data Recording
- Multiplexing / Demultiplexing

PURPOSE-BUILD SwDS

- Satellite Telemetry and Commanding
- Modems and Communications
- Signal Digitization and Transport
- Encryption Interfaces
- Data Acquisition and Analysis

ABOUT SOFTLINK

SOFTLINK, developed by ARKA, provides a versatile open architecture that allows for customized solution-building through its toolset and library of modular, scalable SwDs.
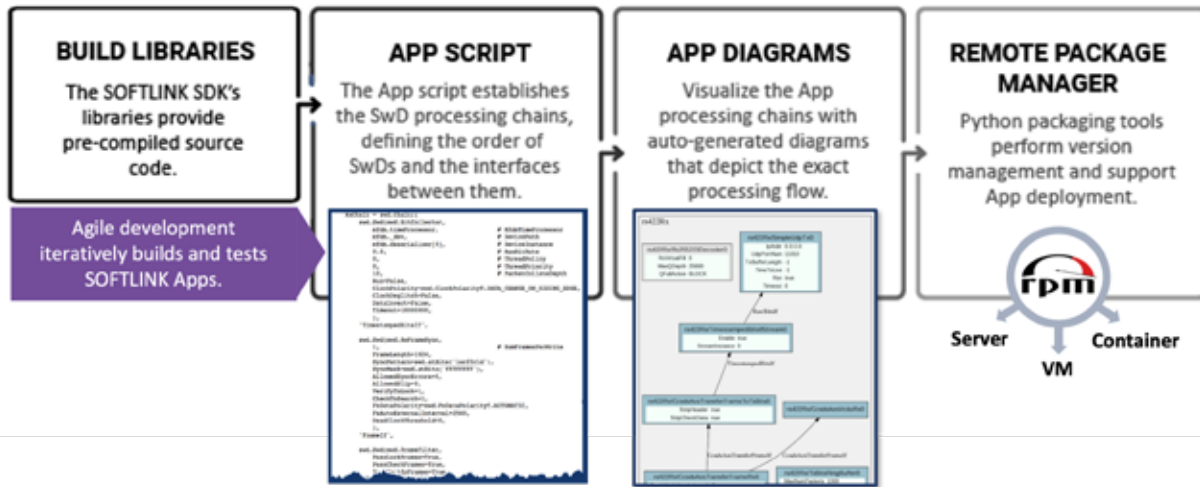
SOFTLINK modules are used in over 2,500 deployed applications.

The SOFTLINK Codebase offers more than 600 SwDs at TRL-9 and an additional 800 reusable and customizable SwDs.

CREATING AN APP WITH SOFTLINK SDK

The SOFTLINK Software App Development Kit (ADK) is used to combine and order SwDs to develop functional end-to-end applications. SOFTLINK has been used to create software for satellite communications, network and data protocol processing, digital twins, and real-time data acquisition.

FOUR STEPS TO CREATE AN APP

BUILD LIBRARIES
The SOFTLINK SDK's libraries provide pre-compiled source code.

Agile development iteratively builds and tests SOFTLINK Apps.

APP SCRIPT
The App script establishes the SwD processing chains, defining the order of SwDs and the interfaces between them.

APP DIAGRAMS
Visualize the App processing chains with auto-generated diagrams that depict the exact processing flow.

REMOTE PACKAGE MANAGER
Python packaging tools perform version management and support App deployment.

Server   Container
VM

APPS CONTAIN MULTIPLE PROCESSING CHAINS OF INTERCONNECTED SwDS

- SOFTLINK follows a functional programming architecture. Data moves along the processing chain from SwD to SwD using nested function calls and memory pointers to achieve high performance and throughput.

- The Python App script links and orders the SwDs and sets up their default parameters.

- A robust suite of tightly controlled C++ data structures ensures interoperability and precise data transfer between SwDs.

- SwDs handle threading, queuing, switching, and routing to support complex data flows.

- SOFTLINK App diagrams are auto-generated to provide a visual reference and make it easy to understand what SwDs are employed, how they are configured, and how data flows through each processing chain.
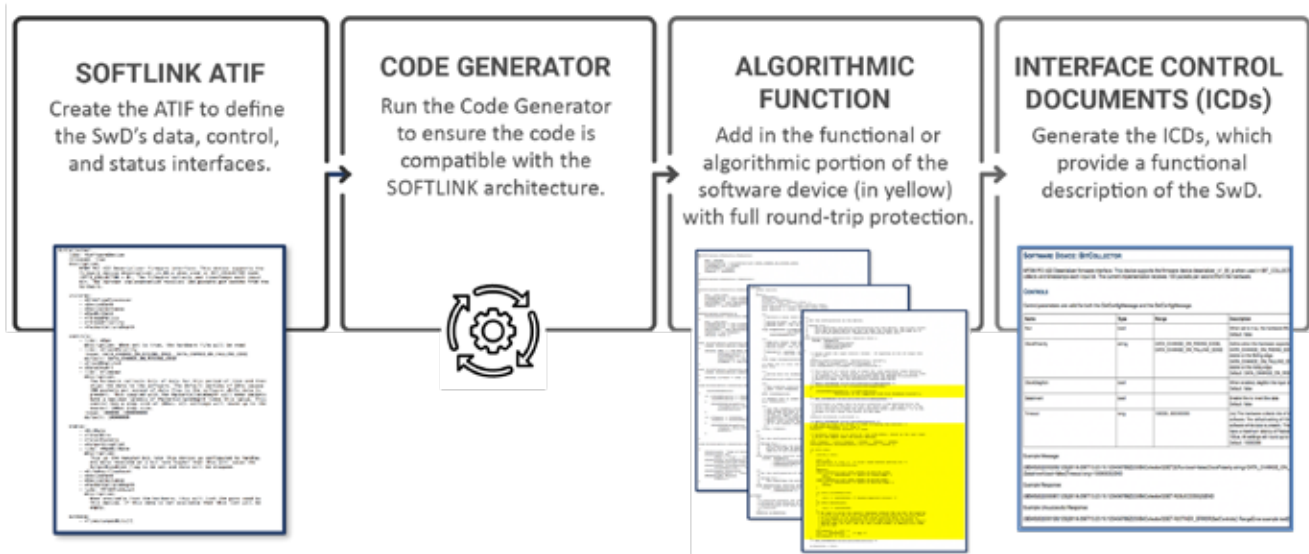

A SwD chain orders and connects SwDs to process a data stream from input to output.

BUILDING NEW SwDS WITH SOFTLINK SDK+

SOFTLINK leverages robust interfaces to connect with user-developed and third-party source code. However, if your application requires development of new SwDs, SOFTLINK SDK+ Code Generators simplify the task.

The use of this technology boosts the reliability and sustainability of the modular software building blocks. SwDs are generated from a YAML-based ARKA Interface File (ATIF), which defines all of the SwD's data transfer interfaces, controls, and status parameters.

FOUR STEPS TO CREATE A SwD

SOFTLINK ATIF
Create the ATIF to define the SwD's data, control, and status interfaces.

CODE GENERATOR
Run the Code Generator to ensure the code is compatible with the SOFTLINK architecture.

ALGORITHMIC FUNCTION
Add in the functional or algorithmic portion of the software device (in yellow) with full round-trip protection.

INTERFACE CONTROL DOCUMENTS (ICDs)
Generate the ICDs, which provide a functional description of the SwD.

BUILDING THE UI FOR SOFTLINK APPS

AMERGINT
User Interface

- System Configs
- App Manager
- TestExec
- inSight
- Docs

SOFTLINK Apps provide RESTful interfaces for management and configuration of the Apps you build and deploy. The SDK user can also build a browser-based UI for the Apps.

The UI Editor provides a user-friendly interface to build and tailor an App-specific user interface by dragging-and-dropping widgets and SwDs onto the Canvas. Fully configurable SwD properties display on the Canvas so that you can customize the SwDs, order them, test, and deploy the App.

UI FEATURES

- Intuitive UI Editor provides the building blocks to develop Apps.
- Auto-generated UI is available for SwD Controls and Status Parameters.
- Test and debug Apps with Processing Chain Navigation, which displays a clear, visual depiction of each processing chain. Move forward and backward within a chain while the UI shows the control and status parameters of each SwD in the chain along with its two neighboring SwDs.

THE SOFTLINK SDK SIMPLIFIES APP DEVELOPMENT

AGILE DESIGN AND DEVELOPMENT

Build SwDs, create applications, and deploy them in any environment: Cloud, virtual, or server-hosted. The SDK is designed to quickly iterate and build software solutions that meet changing needs.

CODE GENERATION TOOLS

Develop new SwDs using sophisticated tools that automate coding of the input, output, control, and status processing for the new SwD. Add in the algorithmic function of the SwD with full round-trip protection as interfaces change.

EXISTING SOFTWARE CONNECTIONS

Use robust software and network interfaces to intuitively connect to existing application software and incorporate third-party libraries.

FUNCTIONAL PROGRAMMING

Use SOFTLINK's inherent functional programming architecture in which each SwD accepts a time-slice of input data, performs a function on that data, and generates a time-slice of output data. These slices of data are continuously passed through the App from SwD to SwD up or down the processing chains.

PRE-COMPILED C++ LIBRARIES

Access over 1,600 tested and reusable SwDs. These SwDs provide a mature (TRL 9) software baseline for new applications.

REDHAT PACKAGE MANAGER (RPM)

Simplify version management with Python packaging tools and RPMs. Install, uninstall, and update single applications across platforms. Roll versions forward and backward as needed.

RESTful APIs

Control and monitor the App using RESTful APIs.

ROBUST TOOLSET AND CODEBASE

Leverage the SOFTLINK toolset and codebase to create your own applications. Optimized for signal, network, and data communications, the SOFTLINK SDK has been used by ARKA for more than a decade.

SIMPLIFIED BUILD | MAKE

Employ distributed development and make piecemeal updates without operational impact. The SDK provides the CMake family of tools as well as the Python and C++ libraries, applications, and other software components that simplify the build|make process for your App design.

Can we help? ARKA's expert team is available to assist in developing Apps to achieve your specific mission or program goals.
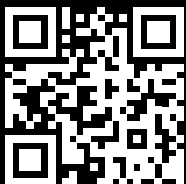
FOR ADDITIONAL INFORMATION:

2315 Briargate Pkwy., Suite 100
Colorado Springs, CO 80920 USA

Tel: 719-522-2800 | Fax: 719-522-2810

in arka-group-technologies   🌐 www.arka.org

ARKA