embenatics
foundation layer software

WHITE PAPER

embenatics Design Methodology

This document is an introduction into the embenatics design methodology and its product portfolio. We offer an embedded software foundation layer (mbLay) that provides all basic operating system services and a well designed communication infrastructure. The foundation layer is available for several operating systems as support for designing and validating portable software applications. Its use is further empowered by a set of additional inbuilt services that support the efficient diagnosis and testing of embedded system software.

The foundation layer is accompanied by a PC based tool suite that assists you in creating the documents required during the embenatics development workflow. embenatics offers an editor to create system resource and interface description documents, and a generator to compile these description documents into several output files which will be used throughout the development and diagnosis phase. A logging and diagnosis tool provides insight into the running software system.

Our design methodology and product portfolio support the design of portable and maintainable software as well as their efficient testing and diagnosis.

The Challenges of Embedded Software Development

One of the challenges in embedded software design is to ensure the portability of a software application. The goal is to avoid modifications in your software application while exchanging the operation system and/or the hardware platform. This becomes more crucial in the age of distributed systems where parts of the application may be moved around among the subsystems of a multi-core platform.

Another challenge is maintainability. Everybody knows how painful it is to change interfaces or even the entire system configuration once you have reached a certain stage in your development process.

Have you ever thought about logging the communication inside an embedded application? How important is it to obtain more information about the inside workings of your application while it is running in the embedded target?

Our focus is on guiding you through this difficult area of embedded software development. The embenatics design methodology is based on a centralized definition of system immanent properties like communication interfaces, system resources and the distribution of threads among the subsystems of your application.
The development environment for a project using the embenatics tool suite and foundation layer is shown in the following diagram:

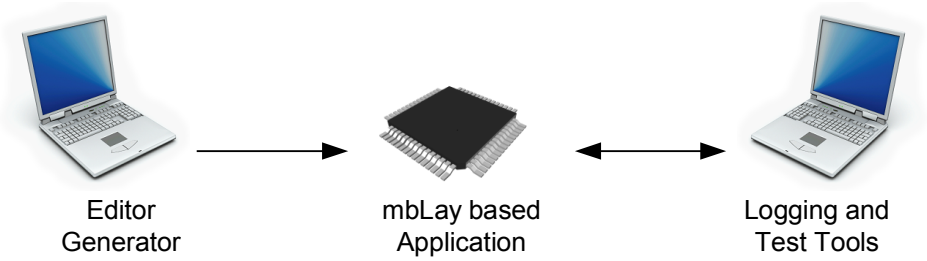

Figure 1 Development Environment

Description Documents are the Base for Software Design

The description documents created in the system design phase are the key input during the development, diagnosis and testing. The following documents are mandatory in an embenatics based project:

- The platform description document contains the platform-dependent size and alignment information of the base data types used in the embenatics tool chain.
- The system description documents provide the configuration of system resources (threads, memory, synchronization ...) and their properties as well as the communication relationship among application threads for a project.
- The interface description documents are used to specify inter-process communication services, data types and definitions for a project.

The description documents are created and maintained with the help of the embenatics editor mbEdit. This PC based editor provides a convenient way to configure the system resources as well as to design the inter-process communication services, data types and definitions. The white papers [embenatics System Description] and [embenatics Interface Description] give detailed introductions into the structure of the description documents and the usage of the editor.

The same description documents are the basis for the diagnosis and test phase to visualize system communication and data structures, as well as to define test vectors and patterns.

Generator mbGen

The generator mbGen is the central generator tool of the embenatics tool chain. All input documents are converted by mbGen to a set of output files which will be used during the development and diagnosis phase. The generator converts all description documents into the following source code templates, C-header and configuration files:

- The system resource configuration that contains tables of the system resources including their properties. These are used by the foundation layer to automatically create all system resources at startup and to boot up the application.
- The system communication configuration that contains tables for the inter-thread and multi-core communication.
- The API prototypes and data type definitions for remote service based communication (RPC).
- The serialization description for the exchange of data structures between threads and on the test and debug interface.

The generation of various products for development, diagnosis and testing from the same description documents ensures consistency of data throughout the entire system development phase - single-source-principle.

embenatics Development Workflow

The following picture shows an overview of the embenatics typical workflow for design, development and diagnosis as it was introduced in the previous paragraphs.

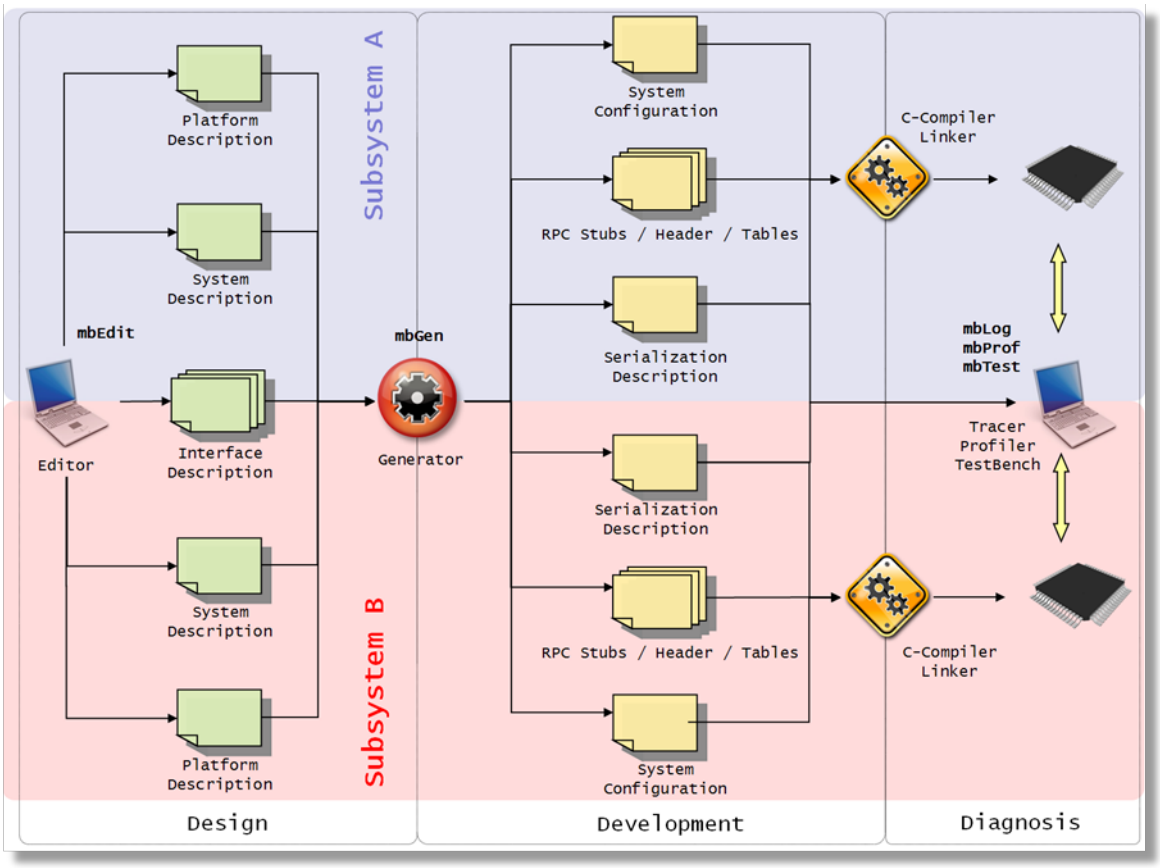

Figure 2 embenatics Workflow

Figure 2 displays the workflow for two subsystems A and B of a distributed application. Starting on the left side the designer utilizes the editor mbEdit to create the platform, system and interface description documents. The description documents are compiled by mbGen and a set of output files is created which will be used during the development and diagnosis phase.

Embedded Foundation Layer - mbLay

The embenatics software foundation layer mbLay is easily added as an overlay to your preferred operating system. It provides solutions for most common requirements that exist in your application development cycle. These are operating system independency, diagnostic capability, testability and the distribution of interacting software components in homogeneous and heterogeneous systems, especially in multi-core environments. The foundation layer is available for several real-time operating systems designed for embedded systems like ThreadX™, VxWorks™, Nucleus™ or FreeRTOS™, as well as for high level operating systems like Windows™, QNX™ or Linux™ and can be easily adapted to further operating systems. The foundation layer collects various statistical and historical data about the status and performance of the threads, allocation of memory and other system resources. This will support you during system optimization and assist you in solving system issues like deadlocks and memory leaks. For further information on the foundation layer please see another document of this series of whitepapers [mbLay Architecture and Interface].

Communication Model

The embenatics communication model allows interaction between software components regardless of the subsystem in which they reside. Due to the centralized system and interface description approach, the communicating peers interact in a transparent way. Therefore, they do not need to know if they are located on the same core, on different cores of the same SoC, or on cores of different systems that are connected by any kind of wide area network. For further information on the communication model please see another document of this series of whitepapers [mbLay Inter-Process Communication Model].

Logging

The embedded foundation layer mbLay provides detailed diagnostic information about the internals of the system, especially the inter-process communication. In addition, mbLay provides a logging API that can be used by application developers to instrument the code in order to output information needed for debugging purposes. For further information on mbLay logging please see the other document of this series of whitepapers [mbLay Logging].
The embenatics tool suite includes an invaluable diagnostic tool for target data logging and visualization to analyze the runtime behavior of the system and your application. All data that is output by the software application is logged and displayed by mbLog. For further information on the logging tool please see the other document of this series of whitepapers [mbLog Logging and Diagnosis Tool].

Conclusion

Embenatics offers a software foundation layer and a tool suite that supports your development team in designing your software in an efficient, portable and maintainable way. The foundation layer hides all hardware and operating system specifics and therefore keeps your implementation independent of the used platform. Our design methodology is based on a centralized description of the system resources and communication interfaces. This approach is the basis of our RPC based communication model. In addition it supports the maintainability of your application and ensures consistency of data used during development, diagnosis and testing.

About Us

embenatics is a new company that entered the market in 2010. Our focus is on embedded software development; as such we offer a software foundation layer and tool suite that supports your development team in designing embedded software in an efficient, portable and maintainable way. Based on our wide and varied experience in embedded systems design and development, we know that future product requirements are hard to predict. Our goal is, therefore, to provide you with our technology to make the design of your products as flexible and adaptable as possible. Our approach allows your company to concentrate on the core competencies that differentiate your valuable product from those of your competitors.

Before embenatics was founded, we worked with well-known international companies over two decades and gained valuable experience in the embedded software business. While working as software developers and architects, we encountered the various challenges of the embedded software development life cycle. This wide range of experiences is the backbone of the software foundation products that are offered by embenatics.

Our business philosophy is to establish a close and trustful relationship with our customers in order to successfully promote and support projects over a long time period. For further information please contact

Joachim Pilz
Beerenstraße 29
14163 Berlin

info@embenatics.com
www.embenatics.com

Phone +49 30 26 34 75 28
Mobile +49 176 96 98 46 07