INNOVATIVE TECHNOLOGICAL SOLUTIONS FROM KM3NeT

Credit: Italiana d-Arte

KM3NeT is a research infrastructure housing the next generation neutrino detectors, located at some of the greatest depths of the Mediterranean sea. Data to and from detector components for monitoring and management purposes are transmitted using a custom designed protocol. KM3NeT uses Web interfaces developed on top of HTTP(S) and using browsers as ubiquitous resources.

KM3NeT uses in-house made Web server libraries to simplify the development of Graphical User Interfaces for the Detector and Acquisition Control system (Control Unit) and for the central Data Base Web Application Server.

The application program does not suffer from typical constraints posed by hosting in a Web server process. The HTTP(S) interface is a thin library (36 KiB) that can be incorporated also into existing applications. Overriding one virtual method is enough to implement the Web interface. While the library is feature-rich, the simplicity of code and the lack-by-design of dangerous functions (e.g. page upload) reduces the attack surface and simplifies secure application development.

Application
No constraints on CPU, memory usage, disk, DB, libraries, etc.
Web Server

HTTP header control
Light (36 KB)
Binary files
AJAX/JSON
.NET/Mono
Protection
SSL/TLS
REST-like interface
Chunked transfers
Session tracking
File upload

KM3NeT Opens a new window on our universe

www.km3net.org, kt@km3net.de

SAWI: Server Application with Web Interface

The software that controls detector operation and data acquisition in KM3NeT (Control Unit) runs as separate processes possibly hosted on different machines and interconnected by a remote procedure call network protocol. Given the availability of in-house Web access instead of using a different messaging protocol, KM3NeT developed its own remote procedure call technique, named Server Application with Web Interface (SAWI) that uses HTTP(S) as transport protocol.

```
public static void ZeroAlgo(string A, int B){
    /* code */
}

public static void OneAlgo(string A, int B){
    /* code */
}

[WSrvPublish]
public static double GreatAlgo(string A, int B){
    /* server algorithm code */
}

public static void OtherAlgo(string A, int B){
    /* code */
}
```

The developer who wants to incorporate SAWI as a library is only required to tag methods to be made publicly available for access through HTTP(S) with a specific attribute. A virtual directory allows browsing the set of published methods and calling them, interactively as well as programmatically.

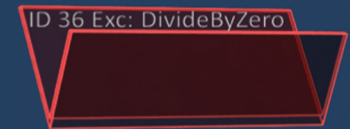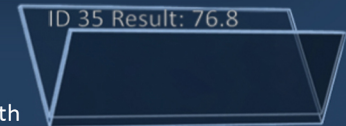Interactive method call through automatically generated HTML pages

WebService page for NotifyT0SetChange – Process Instance 18292
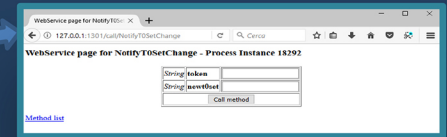
Short calls: return immediately

Long calls are jobs with result persistence

RPC client over HTTP Same library!

Server exceptions
Protocol exceptions

ID 35 Result: 76.8

ID 36 Exc: DivideByZero

Support for long-duration calls is natively included, with job tracking and call result persistence.

The SAWI library is extremely lightweight (less than 20 KiB) and suitable for usage under Mono or .NET/.NET Core (Linux, Windows, MacOS).