

Universal API Interface

The concept of the universal API was introduced in the chapter [Universal API for Access to D2000](#). Before we will describe individual REST and Comet API interfaces, it is necessary, in the following chapter, to describe a way of serializing values and parameters which is similar in both interfaces.

Usability of Individual API Interfaces

To answer the question of why SmartWeb implements two and not just one API interface, we need to look at the appropriateness of individual interfaces used in various situations.

D2000 is a real-time SCADA system that is able to record changes of a high number of, for example, I/O tags. In order for a web or other application to be notified about these changes, it is necessary to implement a communication technology of the [server-push](#) type in which a server alone notifies the client about the change of the object value. Otherwise, it is necessary for the client to periodically ask a server in some time interval whether there were any changes in observed objects. This communication method is, however, very inconvenient. When having a longer time interval, a situation may arise that the object value will change right after the server's answer and the client will not be informed about this value until checking the server again. On the other hand, if the time interval is too short and the value is changing sporadically, considerable wasting of communication means may happen unnecessarily. Because of this, there are situations when we really use a communication technology that supports the server-push. However, the server-push way of communicating is not standardized that is why we needed to choose one existing and available technology. Ideally such communication that would support robust server-push communication between a web application in the browser and a server. After analysing all options, we chose technology based on the [Comet communication concept](#) implemented in the robust JAVA library [Cometd](#) with long production history [and with the use in a cloud solution with more than 150 000 clients](#).

SmartWeb implements the Comet communication concept mostly for acquiring current values and the possibility to [call from D2000 also remote RPC methods](#) registered and performed on a client. Since these usage cases are not always required for the application, it is possible to use easy-implementation and standardized REST API interface that is a subset of the Comet API.

Authentication in Individual API Interfaces

As we have already mentioned in the chapter [Other Functions of the SmartWeb Platform](#), the authentication way of individual interfaces differs. The REST API with its [HTTP-BASIC](#) authentication is more aimed at communication for non-web clients, the Comet API will redirect the unidentified user to the login form (FORM authentication) and that is why this interface is currently naturally suitable for communication between web applications and servers.



In a case of need, also another client can communicate as a web application via the Comet API. The only condition is for the client to be able to authenticate by sending a special HTTP requirement emulating login through a login form. The alternative, however a more complicated possibility, is to directly show the login form to a user in a special window with an embedded browser and after a successful login, extract authentication cookie for another use in the Comet API.