getServiceContext

Syntax	`anArray = getServiceContext(protocol)`
Semantics	Returns an array of **ContextElement** (**Base Components.Basic Behavior. ServiceContext**) objects containing all basic context parameters of the specified protocol related to the current request. If the protocol is not known the operation will return an empty array. For performance considerations, body for HTTP, and header and body for SOAP are excluded from `getServiceContext()`. Use getServiceContextValue() to get these values.
Substitutables	`protocol` A **String** specifying the protocol, of which parameters are requested.

Examples	`set serviceContext = getServiceContext("tcpip");`
	`set param1.name = serviceContext[0].name;`
	`set param2.value = serviceContext[0].value;`

Supported Protocols per Service Type

Service Type / Context	Protocols
All	- System - SystemEnvironment
HTTP	- TCPIP - HTTP
SOAP	- TCPIP - HTTP - SOAP
REST	- TCPIP - HTTP
Persistent State	- Object

Supported Parameters

Protocol	Parameter	Description
TCPIP	RemoteIPAddress	IP address of host that initiated the request. If a proxy is used, this is the IP of the proxy server.
	RemotePort	Port number, from which the request was initiated.
	RemoteHostName	Name of the host that initiated the request. If not resolvable, the IP address is returned.
	LocalIPAddress	IP address of the machine this service runs on.
	LocalPort	Port number of this service.
	LocalHostName	Name of the host this service runs on. If not resolvable, the IP address is returned.

HTTP	Method	HTTP method used to invoke this service. Usually, it returns **POST**.
	URI	URI called to invoke this service.
	Body	Complete content of the HTTP request invoking this service.
	<header>	Any other name will return the value of the HTTP header of the same name, if it exists.
SOAP	Header	Complete source of the SOAP header sent to invoke this service.
	Body	Complete source of the SOAP body sent to invoke this service.
	fault	Content of the SOAP fault if a request comes with a fault element.
System	Host	Local host name. This call returns a value for all service types.
	IP	IP address of the local host. This call returns a value for all service types.
	User	The operating system user who is executing the current service. This user has nothing to do with the external user who might invoke a service. Thus, this call returns a value for all service types.
SystemEnvironment	<environment variable name>	Value of the operating system environment variable. This call returns a value for all service types. Note: the accessible environment variables can be inspected in the **start log** of the current service. ⓘ Since Linux is case sensitive, you should specify all environment variables as they are.
Object	class	The name of the persistent state class.
	ID	The unique technical ID of the of the object as stored to the persistent state database.