



DelphiDay

italian conference

WiRL 4.7

All the new features



PAOLO ROSSI

WINTech ITALIA - CTO

SENCHA & EMB. MVP



 blog.paolorossi.net

 paolo@paolorossi.net

 twitter.com/awebguy

 github.com/paolo-rossi

 linkedin.com/in/paolo-rossi-pc



DelphiDay
italian conference

19-20 Giugno 2025
Piacenza



wintech
italia



LUCA MINUTI



lucominuti.it



luca.minuti@gmail.com



dev.to/lminuti



github.com/lminuti



www.linkedin.com/in/lucominuti



DelphiDay

italian conference

19-20 Giugno 2025
Piacenza



wintech
italia



GITHUB PROJECTS



github.com/paolo-rossi



Delphi JWT

JSON Web Token Library



WiRL

REST Library for Delphi



Linux Daemon

Real Linux daemons



Delphi Neon

JSON Serialization Library



OpenAPI-Delphi

OpenAPI 3.0 Library



NATS Delphi

NATS Client Library for Delphi



AGENDA

1. REST & WiRL
2. Main WiRL features
3. WiRL new features
4. Where (and how) to use the new features



REST & WiRL

1



REST

- Resources
- Content negotiation
- HTTP verbs
- HTTP error codes
- HTTP parameters
 - Query
 - Param
 - Header
 - Body



Features

2



WIRL FEATURES: CORE

- Resource Classes
- Pervasive Content Negotiation
- Attribute-based
- HTTP Methods Mapping
- Entity Providers (Message Body Readers & Writers)
- Powerful JSON support through Neon
- Powerful Client API
- Versatile Filters



WIRL FEATURES: SERVER (1)

- Automatic OpenAPI Documentation
- Role-based Authorization
- Pluggable Authentication
 - Full JWT support
- Pluggable Configuration (fluent style)
- Data Converters
- Content Validation (attribute-based)
- HTTP Parameter Management (attribute-based)



WIRL FEATURES: SERVER (2)

- Attribute-based Endpoint Redirection
- Context Injection with Custom Classes Support
- Automatic Memory Management
 - For parameters and resource results
- Request-based Garbage Collection
- CORS Support and Configuration
- Linux 64bit Support
- Automatic Linux Daemon Generation



WIRL FEATURES: SERVER (3)

- Multiple Server Application Types Supported
- Support for any Database (included for FireDAC and UniDAC)
- GraphQL Support
- Easy Exception Handling
- Multi-Engine: REST, Webserver, Proxy
- Multi-App: Separation Between Resource Groups (Security)
- Chunked Response
- SSE (Server-Sent Events)



WIRL FEATURES: CLIENT

- Idiomatic Entity Requests
- Independence from the HTTP Client Library (Indy, NetHttp)
- Fluent-style Requests
- Design-Time Component Editors for Easy Request Configuration
- Multi-response through Tuples



New Features

3



WIRL NEW FEATURES (4.6)

- SinkPaths config to manage no-response paths (like favicon.ico, etc...)
- Support for "multipart/form-data" in TWiRLClient
- Experimental [SingleRecord] to get a single record (instead of an array) from a dataset
- Ability to retrieve the response content as an object (Response.Content.AsType)



WIRL NEW FEATURES (4.7)

- ProxyServer support
- Engine refactoring: new TWiRLWebServerEngine
- Support for UniDAC database framework
- WebStencils support (and Demo)
- Support for chunked transfers
- Support for SSE (Server Sent Events)



WIRL NEW FEATURES (4.7)

- Container-ready console applications
 - Docker
 - Podman
 - LXC
 - ...
- Official documentation

wirl.delphiblocks.dev



Where & How

4



PROXY ENGINE

- CORS problems?
 - WiRL CORS configuration
 - Use an external WebServer with reverse proxy support
 - Configure a Proxy Engine in WiRL
- Mocking data transfers
- Basic URL Rewriting
- Hiding the server's IP from the client



WEBSERVER ENGINE

- Mapping of directories as paths
- Useful in front-end programming
- Mapping different libraries/assets under the root “/”
- Improves security
- Improves versatility



WEBSTENCILS SUPPORT

- Building of static or hydrated websites
- Easy to use
- Integration with Delphi



CHUNKED TRANSFERS

- LLM/RAG applications
- Streaming-like applications
- A lot of data
 - You want/can work on the data as it arrives



SSE (SERVER SENT EVENTS)

- When polling is too expensive
 - Too many clients
 - You need the data asap
 - (but) The interval for updates is very long
- Beware that SSE requires an open TCP connection to work
 - Impact on the server when clients growth
- SSE has a limitation to the maximum number of open connections
 - Max 6 connections per browser / per server



THANK YOU