

# PHP ZABBIXAPI

a powerful library for the Zabbix JSON-RPC API



who we are

and what we're doing



# who we are



- confirm IT solutions GmbH
- located in Switzerland (primarily german speaking area)
- we're a small & dynamic team
- strong knowledge in
  - UNIX & Linux derivates
  - networking
  - security (i.e. OS hardening, network security)
  - automation (scripting & programming)



# the Zabbix JSON API

a powerful JSON-RPC API



# the Zabbix JSON API



- invented in Zabbix 1.8
- extended in Zabbix 2.0
- API is great to automate changes in Zabbix (external by a script)
- interface for other systems & 3rd party tools
- IMHO a kick-ass functionality compared to other monitoring solutions
- there are no libraries / classes included within Zabbix
- though there are some open-source libraries available (e.g. zbxapi)



# the idea

how we improve the usability of the JSON API



# the idea



- started with a procedural script communicating with the API
- fascinated by the power of the API, the requirements to our script expanded rapidly
- start over with a clean OOP design
- build a library to communicate with the API
- generic request library vs. full implemented library
- why not generate it automatically - is it even possible?
- keep it stupid, keep it simple



# the builder

a powerful utility to generate libraries





# the builder



- compiles the PhpZabbixLibrary (*I'll come back to that the next slide*)
- generates the library directly from the Zabbix 2.0 PHP front-end sources (classes)
- based on templates
- builder is written in PHP
- can compile libraries for “nearly every” scripting / programming language



# the library

an easy-to-use PHP library



# the library



- written in PHP
- built library stored in an abstract class (updatable)
- user-defined customisations stored in a concrete class (empty by default)
- “real implementation” of each Zabbix API method
- SPOC for every API method (*I'll come back to that later*)
- throws Exceptions
- IDE friendly



# a quick demonstration

let's see how it (hopefully) works



# some caveats

the “dark side” of the API



# some caveats



- Zabbix will modify / delete an API method
  - the **bad** thing: the library get's "broken"
  - the **good** thing: existing builder can build a new compatible stock library
  - *either* you've to update each code occurrence in your implementation
  - *or* you've to implement a "deprecated" method / workaround in your concrete API class
- Zabbix will change the procedure, how API methods are initiated
  - the **bad** thing: the builder get's "broken"
  - the **good** thing: the existing library will still work
- But: Usually you can easily update the API, and since you've an abstract implementation with your customisations, you don't have to care about patches and the like



# our roadmap

what's up next



# our roadmap



- keep library up-to-date with new Zabbix versions
- make library available for other programming languages
- other Zabbix & Zabbix API related tools already under development
  - versioned configuration database (CMDB) for servers (e.g. NICs, VLANs, installed LPP/RPM packages)
  - automated accounting for virtual servers (vmware, Citrix, KVM, IBM LPARs) based on resource consumption (i.e. memory, CPU, network & storage)
- everything released under GPL





thank you for listening...

questions?

download the PhpZabbixApi at <http://zabbixapi.confirm.ch>

**confirm IT solutions GmbH**  
Rathausstrasse 14 | CH-6340 Baar | [www.confirm.ch](http://www.confirm.ch)

