# Adding "ArrayOfArrayOf" to Zend_Soap_AutoDiscover - Renan de Lima

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

## Zend Framework: Zend_Soap_AutoDiscover Component Proposal

| | |
|---|---|
| **Proposed Component Name** | Zend_Soap_AutoDiscover |
| **Developer Notes** | http://framework.zend.com/wiki/display/ZFDEV/Zend_Soap_AutoDiscover |
| **Proposers** | Renan de Lima |
| **Zend Liaison** | TBD |
| **Revision** | 1.0 - 11 May 2009: Initial Draft. (wiki revision: 4) |

## Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
9. Class Skeletons

## 1. Overview

Adding complex array structure to WSDL generated by Zend_Soap_Wsdl and Zend_Soap_AutoDiscover.

Some people use WSDL as the webservice API documentation. WSDL that shows just "array" as input or output request doesn't help the client side team. This proposal may provide additional information to WSDL when server side wants to work with array. The major goal is to answer the commom question: what does this array contain? an integer? an string?

## 2. References

- http://www.w3.org/2002/ws/databinding/edcopy/advanced/advanced.html#pattern-SOAPEncodedArray
- http://www.w3.org/2002/ws/databinding/examples/6/09/SOAPEncodedArray/
- http://www.w3.org/TR/wsdl#_types

## 3. Component Requirements, Constraints, and Acceptance Criteria

- This proposal must provides an easy way to declare multi-dimensional arrays in webservices classes
- Current Zend_Soap_AutoDiscover and Zend_Soap_Wsdl behavior still must work for "@param array" and "@return array"

## 4. Dependencies on Other Framework Components

No dependency.

## 5. Theory of Operation

Users just will have to use new documentation style into their webservices classes.

## 6. Milestones / Tasks

- Milestone 1: design
- Milestone 2: tests
- Milestone 3: documentation

## 7. Class Index

No new class is necessary.

## 8. Use Cases

**UC-01 classic usage**

```
<?php
/**
 * @param integer[]
 * @return void
 */
public function xxx(array $number){}

/**
 * @param array of integer
 * @return void
 */
public function xxx(array $number){}
```

WSDL must contain in "message" part:

```
<message name="xxxRequest">
    <part name="yyy" type="tns:arrayOfInteger"/>
</message>
```

WSDL must contain in "types" part:

```
<xsd:complexType name="ArrayOfInteger">
    <xsd:complexContent>
        <xsd:restriction base="soap-enc:Array">
            <xsd:attribute ref="soap-enc:arrayType" arrayType="tns:integer[]"/>
        </xsd:restriction>
    </xsd:complexContent>
</xsd:complexType>
```

### UC-02 multiple array

```
<?php
/**
 * @param integer[][]
 * @return void
 */
public function xxx(array $number){}

/**
 * @param array of array of integer
 * @return void
 */
public function xxx(array $number){}
```

WSDL must contain in "message" part:

```
<message name="__SOME_MESSAGE_NAME__">
    <part name="__SOME_PART_NAME__" type="tns:ArrayOfArrayOfInteger"/>
</message>
```

WSDL must contain in "types" part:

```
<xsd:complexType name="ArrayOfInteger">
    <xsd:complexContent>
        <xsd:restriction base="soap-enc:Array">
            <xsd:attribute ref="soap-enc:arrayType" arrayType="tns:integer[]"/>
        </xsd:restriction>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="ArrayOfArrayOfInteger">
    <xsd:complexContent>
        <xsd:restriction base="soap-enc:Array">
            <xsd:attribute ref="soap-enc:arrayType" arrayType="tns:ArrayOfInteger[]"/>
        </xsd:restriction>
    </xsd:complexContent>
</xsd:complexType>
```

### UC-03 multiple array together complex type

```
<?php
/**
 * @param MyClass[][]
 * @return void
 */
public function xxx(array $number){}
```

WSDL must contain in "message" part:

```
<message name="__SOME_MESSAGE_NAME__">
    <part name="__SOME_PART_NAME__" type="tns:arrayOfArrayOfMyClass"/>
</message>
```

WSDL must contain in "types" part:

```
<xsd:complexType name="MyClass">
    <xsd:all>
        ...
    </xsd:all>
</xsd:complexType>
<xsd:complexType name="ArrayOfMyClass">
    <xsd:complexContent>
        <xsd:restriction base="soap-enc:Array">
            <xsd:attribute ref="soap-enc:arrayType" arrayType="tns:MyClass[]"/>
        </xsd:restriction>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="ArrayOfArrayOfMyClass">
    <xsd:complexContent>
        <xsd:restriction base="soap-enc:Array">
            <xsd:attribute ref="soap-enc:arrayType" arrayType="tns:arrayOfMyClass[]"/>
        </xsd:restriction>
    </xsd:complexContent>
</xsd:complexType>
```

## 9. Class Skeletons

No new class is necessary, zf core team have to change Zend_Soap_Wsdl and Zend_Soap_AutoDiscover internal methods.

]]></ac:plain-text-body></ac:macro>
]]></ac:plain-text-body></ac:macro>