



December 3-6, 2007, Santa Clara Marriott, Santa Clara, CA

WS-Management on Rails

Klaus Kämpf



Novell®

Mission

Make development of
Management Applications
easy and fun

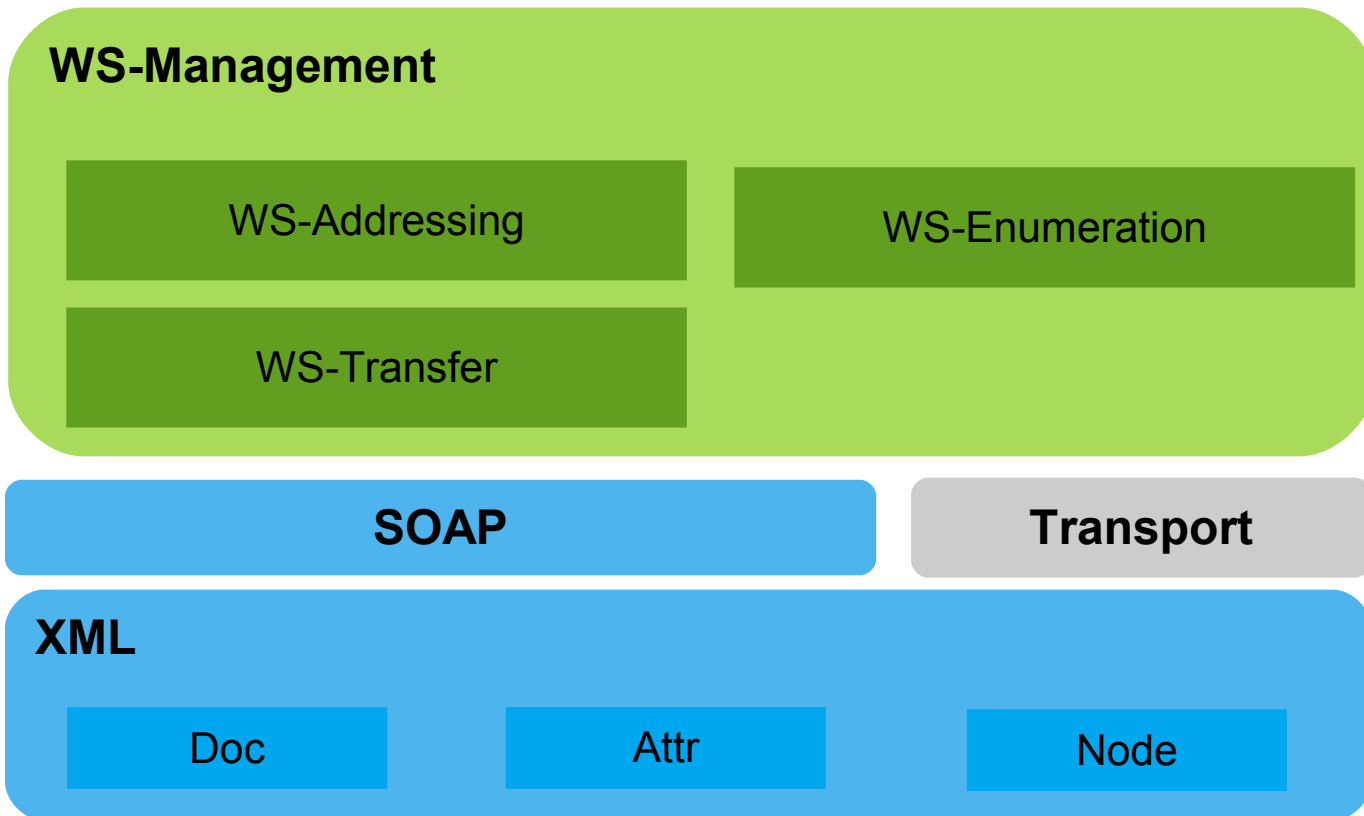
- Openwsman
- Introducing Ruby
- WS-Management with Ruby
- WS-CIM with Ruby
- Rails application

WS-Management

Openwsman

- WS-Management and WS-CIM
- Server and Client
- Open source, portable
- Native API in C
- C++ and Python bindings
- See <http://www.openwsman.org>

Openwsman Client API



Introducing Ruby

Ruby is a

- **dynamic**
 - **object-oriented**
- scripting language with

- a strong focus on simplicity and productivity
- an elegant, natural syntax
- roots in Lisp, Perl and Smalltalk
- easy ways to extend it

- XML Operations
- Transport Operations
- Create, Get, Put, Delete
- Enumerate
- Invoke
- Identify
- Eventing (future)

Stop the *Themes* service in Vista

```
1  require "wsman"
2
3  cimuri = "http://schemas.microsoft.com/wbem/wsman/1/wmi/root/cimv2"
4
5  client = WsMan::Client.new( "http://vista.microsoft.com:80/wsman" )
6
7  options = WsMan::ClientOption.new
8
9  options.selector_add( "Name", "Themes" )
10
11 client.invoke( cimuri+"/Win32_Service", "StopService", options )
```

WS-CIM

CIM_Service Instances Java

```
1  CIMClient cimClient = new CIMClient(
2      new CIMNameSpace("http://localhost", "root/cimv2"),
3      new UserPrincipal(""), new PasswordCredential(""));
4
5  CIMObjectPath op = new CIMObjectPath("CIM_Service");
6  Enumeration instances = cimClient.enumerateInstances(op);
7
8  while (instances.hasMoreElements()) {
9      CIMInstance service = (CIMInstance)instances.nextElement();
10
11     System.out.print(service.getProperty("Name").getValue().toString()
12         + " " +
13         service.getProperty("Started").getValue().toString()
14     );
15
16 }
```

CIM_Service Instances Perl

```
1 use Net::OpenWBEM::Client;
2
3 my $WBEM = Net::OpenWBEM::Client->session(host => "localhost");
4 my $ns = 'root/cimv2';
5 my $instanceEnum = $WBEM->enumInstancesE($ns, 'CIM_Service');
6
7 while ($instanceEnum->hasMoreElements)
8 {
9     my $ci = $instanceEnum->nextElement;
10    print $ci->getStringProperty('Name'), " ";
11    print $ci->getStringProperty('Started'), "\n";
12 }
13 $WBEM->close;
```



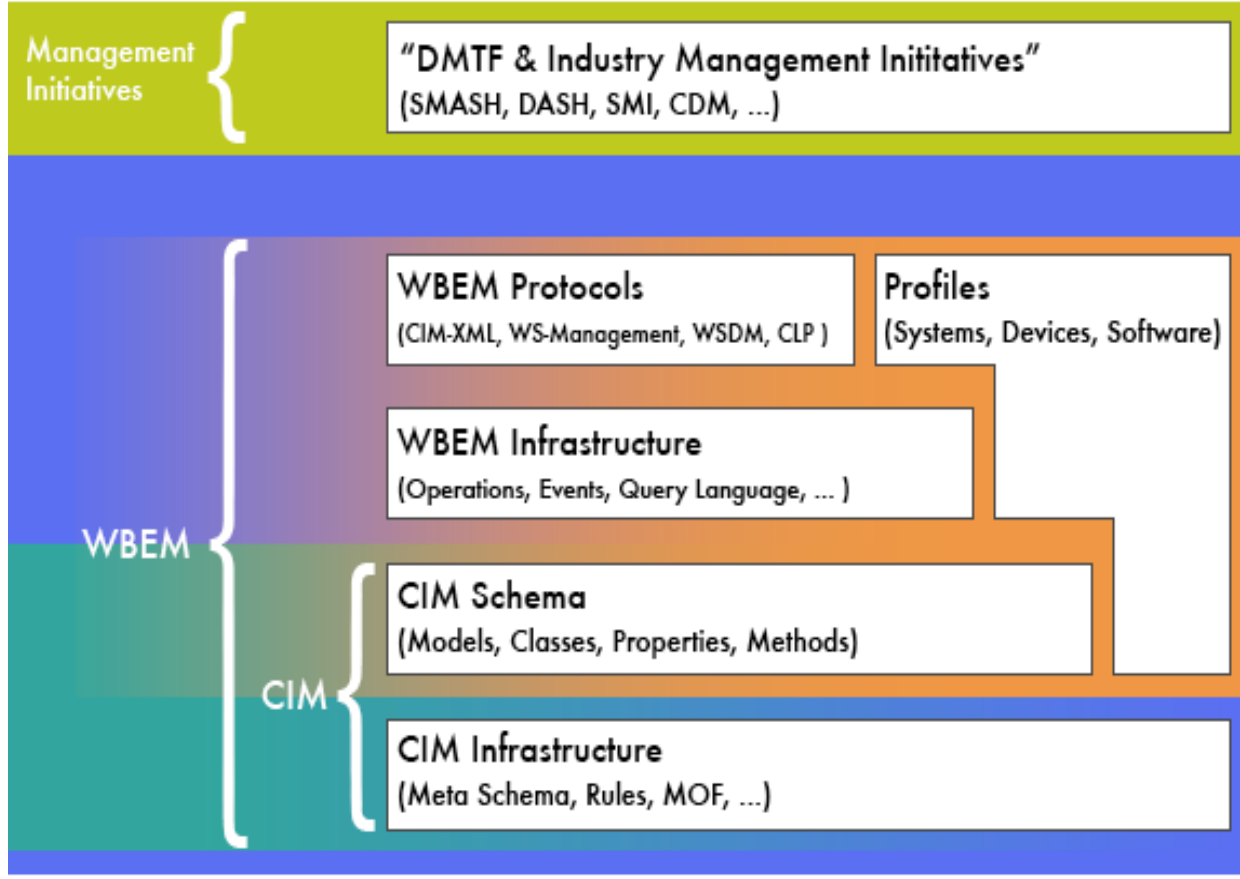
CIM_Service Instances Windows PowerShell

```
1 $colItems = get-wmiobject -class "CIM_Service" `
2 -namespace "root\cimv2" -computername "localhost"
3
4 foreach ($objItem in $colItems) {
5     write-host $objItem.Name, $objItem.Started
6 }
```

CIM_Service Instances Ruby

```
1  require "rwscim"  
2  
3  client = WsMan::Client.new( "http://localhost:8889/wsman")  
4  
5  WsCim::Service.each( client ) { |service|  
6    puts "#{service.Name}, #{service.Started}"  
7  }
```

DMTF WBEM Stack



- Covers the CIM metamodel
 - Feature, Property, Method
 - Instance, Reference, Association
- Intrinsic operations

- Covers the CIM metamodel
 - Feature, Property, Method
 - Instance, Reference, Association
- Intrinsic operations

```
1   require "rwscim"  
2   client = WsMan::Client.new( "http://localhost:8889/wsman")  
3   instance = WsCIM::Service.first  
5   properties = instance.properties  
6   keys = instance.keys  
7   methods = instance.methods  
8   references = instance.references
```

mofgen

- MOF -> C -> Ruby compiler
- Based on CIMPLe mof compiler
- Generates C extensions for Ruby
- CIM Instances as first-class Ruby Objects
- Obsolete with WS-Catalog

- MOF -> C -> Ruby compiler
- Based on CIMPLE mof compiler
- Generates C extensions for Ruby
- CIM Instances as first-class Ruby Objects
- Obsolete with WS-Catalog

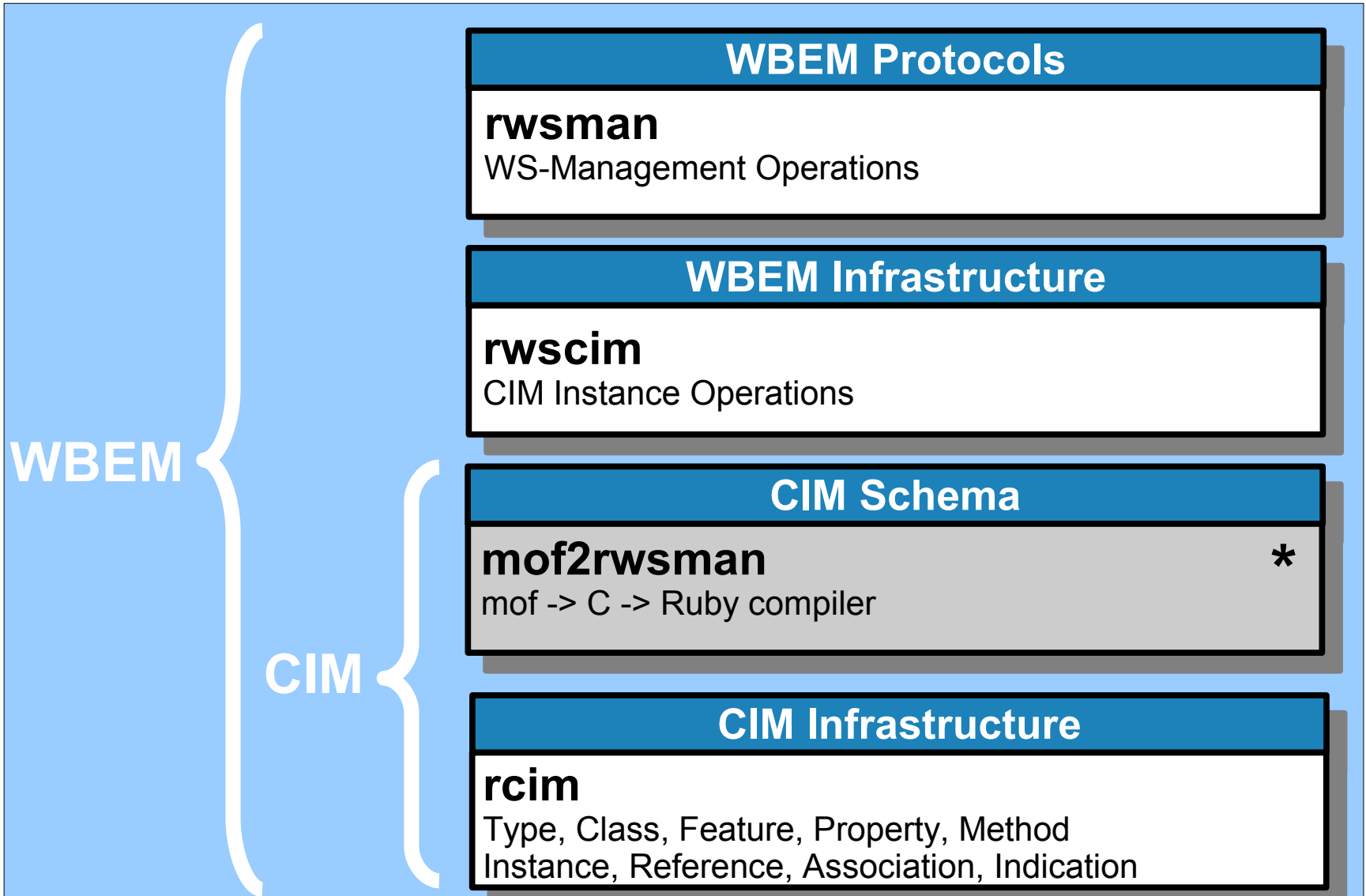
```
> cd cimple/schema/cim214
> ../../bin/mof2rwsman -I . CIM_Schema.mof -c CIM_Service
> ls cim_*
> cim_service.c  cim_service.h
```

- WBEM Protocol stack
- Glues generated Ruby extensions

- WBEM Protocol stack
- Glues generated Ruby extensions

```
1   require "rwscim"
2
3   client = WsMan::Client.new( "http://localhost:8889/wsman")
4
5   WsCIM::Service.each( client ) { |service|
6     puts "#{service.Name}, #{service.Started}"
7   }
```

WBEM with Ruby



Web Management Application

Why Rails ?



About Ruby On Rails

Web development that doesn't hurt

- Programmer happiness
- Productivity
- Convention over configuration
- MVC pattern
- No compilation
- Easy deployment and scaling
- Ajax

WS-Management on Rails


File Edit View History del.icio.us Bookmarks Tools Help

http://localhost:3000/start

Google

WS-Management on Rails

WS-Management and WS-CIM with Ruby



[Home](#) [Users](#) [Hosts](#) [Intro](#) [About](#)

[Logout](#) [Release](#)

Status

User
Openwsman default wsman

Host
localhost 127.0.0.1

Actions

Inventory

Status

Browse

HAL

YaST

Search

Host Identification

Openwsman Project 1.5.9 supporting protocol
<http://schemas.dmtf.org/wbem/wsman/1/wsman.xsd>

Info

This demo shows a Ruby On Rails application utilizing Web Services for Management

Created by Klaus Kämpf

Links

- ◆ [Ruby](#)
- ◆ [Ruby On Rails](#)
- ◆ [DMTF](#)
- ◆ [Openwsman](#)
- ◆ [rwsman - Ruby bindings](#)

Sponsors

- ◆ [Novell](#)
- ◆ [OpenSUSE](#)
- ◆ [Intel](#)

Template design by [Six Shooter Media](#).

© All your copyright information here.

Done

WS-Management on Rails


File Edit View History delicious.us Bookmarks Tools Help

http://localhost:3000/start

Google

WS-Management on Rails

WS-Management and WS-CIM with Ruby



[Home](#) [Users](#) [Hosts](#) [Intro](#) [About](#)

[Logout](#) [Release](#)

Status

User
Openwsman default wsman

Host
localhost 127.0.0.1

Actions

Search

Host Identification

Openwsman Project 1.5.9 supporting protocol
<http://schemas.dmtf.org/wbem/wsman/1/wsman.xsd>

Info

This demo shows a Ruby On Rails application utilizing Web Services for Management

Created by Klaus Kämpf

Links

- ◆ [Ruby](#)
- ◆ [Ruby On Rails](#)
- ◆ [DMTF](#)
- ◆ [Openwsman](#)
- ◆ [rwsman - Ruby bindings](#)

Sponsors

- ◆ [Novell](#)
- ◆ [OpenSUSE](#)
- ◆ [Intel](#)

Template design by [Six Shooter Media](#).
© All your copyright information here.

Done

WS-Management on Rails


File Edit View History del.icio.us Bookmarks Tools Help

http://localhost:3000/view/instances?classid=16

Google

WS-Management on Rails

WS-Management and WS-CIM with Ruby



Home Users Hosts Intro About

Logout Release

Status

User
Openwsman default wsman

Host
localhost 127.0.0.1

Actions

- Inventory
- Status
- Browse
- HAL
- YaST

Search

Instances of PG_UnixProcess on localhost

Instances

[Reconfigure view]

Nr	Name	Priority	UserModeTime	KernelModeTime
1	init	15	0	3000
2	kthreadd	12	0	0
3	migration/0	4294967196	0	0
4	ksoftirqd/0	34	0	0
5	migration/1	4294967196	0	0
6	ksoftirqd/1	34	0	0
7	events/0	10	0	0
8	events/1	10	0	0
9	khelper	10	0	0
10	kblockd/0	10	0	0
11	kblockd/1	10	0	0
12	kacpid	15	0	0
13	kacpi_notify	15	0	0
14	cqueue/0	11	0	0
15	cqueue/1	11	0	0
16	kseriod	10	0	0
17	pdflush	17	0	0
18	pdflush	15	0	0
19	kswapd0	11	0	0
20	aio/0	11	0	0
21	aio/1	11	0	0

Info

This demo shows a Ruby On Rails application utilizing Web Services for Management

Created by Klaus Kämpf

Links

- Ruby
- Ruby On Rails
- DMTF
- Openwsman
- rwsman - Ruby bindings

Sponsors

- Novell
- OpenSUSE
- Intel

Done

Where's the source ?

- Openwsman
<http://www.openwsman.org>
- Ruby bindings for Openwsman
<http://rwsman.rubyforge.org>
- Ruby On Rails
<http://www.rubyonrails.com>
- RPM Packages
<http://en.opensuse.org/User:Kwk>

Whats next ?

- On the fly MOF compilation
- Method invocation
- WS-Eventing
- WS-Catalog
 - Dynamic creation of Ruby classes

That's all folks !

Thank you !

Mail to: kkaempf@suse.de

shell - SBLIM wbemcli

```
1 wbemcli ein http://localhost:5988/root/cimv2:CIM_Service |
2 while read instanceName; do
3     instance=wbemcli gi http://$instanceName `
4     Name=`echo $instance
5     | sed -n 's/^.*,Name="\([^"]*\)",.*$/\1/p' `
6     Started=`echo $instance
7     | sed -n 's/^.*,Started=\([^,]*\),.*$/\1/p' `
8     echo "$Name $Started"
9 done
```