* Helft punten laatste reeks
* 1 vraag powershell op test (moeje in januari niet meer kennen) (theorie)
* Veel active perl documentatie copy paste mogelijk­­­

Reeks0

Via COM server laat OS u toe COM klassen(objecten) te gebruiken in een COM client(vergelijkbaar met API voor software). Via dit functionaliteit raadplegen van toepasingen of OS. COM objecten hebben attributen en methodes. Value/key Info hierover te vinden in regedit

In regedit (HKEY\_CLASSES\_ROOT) zoeken:

* Iets te weten komen over CDO.message. Scroll naar CDO.message in regedit. In subtak hiervan zie je verwijzing naar zijn CLSID map ervoor (via het GUID)
* CLSID: bevat mapjes met alle GUID’s van COM objecten
* Edit-> find: zoeken naar bv Scripting.FileSystemObject ProgId=klassenaam, mapnaam=GUID. In de map drukken op submap inprocServer32 (ddl) of Localserver32 (exe) voor te kijken waar implementatie staat. Al deze shit staat in CLSID map.
* Kijken in een CLSID tak. Zit er een typelib bij? Dit gebruik je voor constanten in het script te bekijken. Met die typelib idee ga je dan in oleview kijken.

Een bepaalde Type library (geidentificeerd door GUID) bekijken 🡪 Oleview: vind ‘Microsoft Excel 12.0 Object Library’: Type library tak dan scrollen naar juiste naam

* Bevat methodes, argumenten, terugkeerwaarde.

MSDN library: documentatie van COM klassen

ActivePerl: veel documentatie in

* ActivePerl FAQ / Using OLE with Perl
* Perl Core Documentation
* Modules / Win32 / OLE (helemaal onderaan)

Objecten spreek je dus aan via interfaces (hebben GUID)

Reeks 1

COM object intiliasieren $cdo = **Win32::OLE->new**("CDO.Message");

Gelukt? print **ref** $cdo;

Extra functie inladen use Win32::OLE **qw**(in with);

Ingelade modules info while (($key,$value)=each **%INC**) {}

Type(s) com object @items = Win32::OLE->**QueryObjectType**($cdo)

Ingeladen OLE objecten (debugging)

$Count = Win32::OLE->**EnumAllObjects**(  
 **sub** {  
 my $Object = **shift**;  
 printf "\n%-30s : %s",ref($Object),join(" / ",Win32::OLE->QueryObjectType($Object));  
 });

Laatste foutmelding Win32::OLE->**LastError()**

Stoppen en errors printen $Win32::OLE::**Warn = 3**;

**Scripting.FileSystemObject: $fso**

ms-help://MS.MSDNQTR.v90.en/script56/html/jsoriFileSystemObjectMethods.htm  
ms-help://MS.MSDNQTR.v90.en/script56/html/jsobjfile.htm

$fso = Win32::OLE->new("Scripting.FileSystemObject");

bestaat file $fso->**FileExists**(x)

file bestand krijgen $fso->**GetFile**(x)

File bevat attributen:Oa. File->**{path}** File->**{type}**

Folder bestand krijgen $fso->**GetFolder**(".");

Is het van bepaald type if ($file->{Type} =~/Excel/)

Alle files in de dir overlopen foreach my $file (**in** $folder->**Files**){}  
In gebruikt: dus in begin zeker qw{in}

## Mailing

ms-help://MS.MSDNQTR.v90.en/cdosys/html/39186eaa-c4c1-430a-9715-35e291925c5c.htm

|  |
| --- |
| **Methode 1**  use Net::SMTP;  $smtp = **Net::SMTP->new('smtp.ugent.be')**; #stel de juiste smtp-server in  $smtp->mail(' @ugent.be');  $smtp->to(' @ugent.be');  $smtp->data();  $smtp->datasend("Subject: testje met smtp\n");  $smtp->datasend("test\n");  $smtp->dataend();  $smtp->quit; |

|  |
| --- |
| **Methode 2**  **Initialiseren message:**  use Win32::OLE;  $mail = Win32::OLE->**new("CDO.Message");**  $mail->{TextBody} = 'Hallo';  $mail->{Subject} = 'COM' ;  $mail->{From} = '...@ugent.be'; #vul correct aan  $mail->{To} = '...@ugent.be'; #vul correct aan  **Configuratie aanpassen:**  my $conf = Win32::OLE->**new('CDO.Configuration');**  **$conf->Fields**("http://schemas.microsoft.com/cdo/configuration/smtpserver")**->{Value} =** "smpt.ugent.be"; #thuis aanpassen  $conf->Fields("http://schemas.microsoft.com/cdo/configuration/smtpserverport") ->{Value} = 25; #niet noodzakelijk  $conf->Fields("http://schemas.microsoft.com/cdo/configuration/sendusing")->{Value} = 2;  **$conf->{Fields}->Update();**  **Niet die lange namespace gebruiken:**  **use Win32::OLE::Const**;  $constanten =Win32::OLE::**Const->Load**($conf);  $smtpServer = **$constanten->{**cdoSMTPServer};  $conf->Fields($smtpServer)->{value} = "smtp.ugent.be";  … |

Reeks 2 Excel

**Bestaand excel proces gebruiken of nieuw en quit uitvoeren als script stopt.**

**$excelAppl = Win32::OLE->GetActiveObject('Excel.Application') || Win32::OLE->new('Excel.Application', 'Quit');**

Visueel tonen $excelAppl->{visible}=1;

Hoeveel workbooks open $excelAppl->{Workbooks}->{Count}

->{Workbooks}->Add()

Geen error over ‘save first’ $excelAppl->{DisplayAlerts}=0;

Niet onmiddelijk Quit Print “Press ENTER to exit”; <STDIN>;

Open a file (rel path!) $book=$excel->{Workbooks}->Open($path);

Make new file:

my $dir = $fso->GetAbsolutePathName(".");

my $path = $dir."\\".$filename;

$book = $excel->{Workbooks}->Add();

$book->SaveAs($path);

Amount of tabs $book->Worksheets->{Count}

First sheet $sheet = $book->Worksheets(1);  
bepaalde $sheet\_advalvas = $book1->{Worksheets}->Item("ad valvas");

Check all sheets in book foreach $nsheet (in $book->{Worksheets}){}

Range waarden overlopen: $mat = $range->{Value}

* Indien >1 range: geeft array ref terug  
   $range->{rows}  
   $range->{columns}  
   join(" \t",@{$\_}),"\n" foreach @{$mat}
* Indien =1 geeft 1 scal value terug
* Indien =0 empty  
   $mat ? scal : empty

Ranges:

$used\_cells\_range = $sheet->UsedRange();

$cell = $nsheet->Range("A1")->SpecialCells(xlCellTypeLastCell);  
$range\_start\_to\_last = $nsheet->Range("A1",$cell);

$hardcoded\_range = $sheet->Range("A1:D10");

$range\_one\_cell = $sheet->Cells($row, $col)

$range\_between\_cells = $sheet->Range($sheet->Cells($row1,$col1),$sheet->Cells($row2, $col2));

Update

Single value: $range->{Value}=20;

Multiple: $mat->[$i][$j]=$value;  
 $range->{Value}=$mat

Stijl:

$range->Rows(1)->{font}->{bold} = 1;

%constanten = %{Win32::OLE::Const->Load($excel)};   
$range->Borders($constanten{xlInsideVertical})->{LineStyle} = $constanten{xlContinuous};  
…  
$range->rows(1)->Borders($constanten{xlEdgeBottom})->{LineStyle} = $constanten{xlContinuous};  
  
$group\_range->BorderAround(6);

Alles genoeg ruimte: $sheet\_advalvas->{UsedRange}->{Columns}->AutoFit();

Bepaalde value uit ne matrix: ${ ${ $mat }[$row] }[$col]

Reeks 3

# Wql

Dubbele aanhalingstekens gebruiken zoveel mogelijk

Meta\_class voor klasdefinities  
[ \_\_THIS ISA gebruiken ]

Where \_\_SUPERCLASS=’’ voor empty field

# Associators

Geen spaties naast {} vanbinnen

Associators OF {CLASS.att=id} Where ClassDefsOnly enkel klassen

Where resultClass = x enkel die klassen

Associators OF {CLASS} Where schemaonly enkel klassen

# Belangrijke Queries

**Interne polling**  
Select \* from win32\_processtrace where ProcessName = “Notepad.exe”  
**zonder interne polling**  
SELECT \* FROM \_\_InstanceOperationEvent Within 10  
WHERE TargetInstance ISA 'Win32\_Process'  
 And TargetInstance.Name = 'notepad.exe'  
 And (\_\_CLASS = '\_\_InstanceCreationEvent'  
 Or \_\_CLASS = '\_\_InstanceDeletionEvent')

SELECT \* FROM \_\_InstanceModificationEvent Within 5

WHERE TargetInstance ISA 'Win32\_Service'

And PreviousInstance.Started = true

And TargetInstance.Started = false

bestand aangemaakt in c:\temp  
SELECT \* FROM \_\_InstanceCreationEvent Within 10  
 WHERE TargetInstance ISA 'CIM\_DataFile'  
 And TargetInstance.Drive = 'c:'  
 And TargetInstance.Path = '\\temp\\'

# Notification queries

* Select \* from \_\_InstanceModificationEvent
* Within X pollingtijd
* Where TargetInstance ISA ‘X’ type event

## Timer maken

\_\_IntervalTimerInstruction object aanmaken

Maakt elke minuut(in ms) \_\_TimerEvent

Select \* from \_\_TimerEvent

# Grouping

Group by TargetInstance.\_\_CLASS WITHIN 60 per minuut events genereren van dat type

Todo vanaf 39

# Reeks 4

## Initialisatie

use Win32::OLE 'in';  
$Win32::OLE::Warn = 3;  
my $ComputerName = "."; #lokaal toestel  
my $NameSpace = "root/cimv2";  
my $Locator=Win32::OLE->new("WbemScripting.SWbemLocator");  
my $WbemServices = $Locator->ConnectServer($ComputerName, $NameSpace);

# klasse

## absoluut pad

my $moniker="winmgmts://$ComputerName/$NameSpace:$ClassName";  
my $Klasse = Win32::OLE->GetObject($moniker);

stoppen indien fout in connectie:  
Win32::OLE->LastError()==0 || die “rip”;

## relatief pad

# bepaalde instantie

## Abs Path

$DeviceID="9";

my $moniker="winmgmts://$ComputerName/$NameSpace:$ClassName.DeviceID=\"$DeviceID\"";

my $Instance = Win32::OLE->GetObject($moniker);

## Rel Path

**my $Instance = $WbemServices->Get("$ClassName.DeviceID=\"$DeviceID\"");**

# Alle instanties

1) my $Instances = $WbemServices->InstancesOf("Win32\_OperatingSystem");

2) my $class = $WbemServices->get("Win32\_OperatingSystem");

my $Instances = $class->Instances\_();

3) my $Instances = $WbemServices->ExecQuery("SELECT \* FROM  
 Win32\_OperatingSystem");

4) alle klassen als instanties: select \* from meta\_class of ->SubclassesOf()

->SubclassesOf($classname, wbemQueryFlagShallow) enkel eerste onmiddelijke niveau

Array van bepaald attribuut van $instances  
@DeviceId=map{$\_->{DeviceID}} in $Instances;

# Associators

Instanties  
my $Associators = $WbemServices->AssociatorsOf("Win32\_Directory", undef,undef,undef,undef,undef,1);

**Of**

Instanties  
my $Associators = $Instance->Associators\_();

Klassen  
my $Associators = $Instance->Associators\_(undef,undef,undef,undef,1);

### Enkel van bepaalde klasse

my $Instances = $WbemServices->AssociatorsOf($relpad , undef,$ClassName);

via query

my $WbemObjectSet = $WbemServices->ExecQuery("Select \* from Win32\_IRQResource WHERE IRQNumber=$IRQNumber");

(my $WbemObject)=(in $WbemObjectSet);

my $Instances = $WbemObject->Associators\_(undef,$ClassName);

# Properties/systeemproperties  SWbemPropertySet

* van alle properties hun name, value checken
* 2de vorm todo: zelf herschreven, nog eens nakijken
* Kijken met ref welk type het is, indien array dan join je het, anders toonjet direct
* Properties\_ en SytemProperties\_ via komma gescheiden zoda je beide looped

printf "%-30s %s\n",

$\_->{Name},   
 ref( $\_->{Value} ) eq "ARRAY" ? join ",",@{$\_->{Value}} : $\_->{Value}

foreach sort {uc($a->{Name}) cmp uc($b->{Name})} in $Instance->{Properties\_},$Instance->{SystemProperties\_};

foreach sort {uc($a->{Name}) cmp uc($b->{Name})} in $Instance->{Properties\_}, $Instance->{SystemProperties\_}{

printf "%-30s %s\n",

$\_->{Name},   
 ref( $\_->{Value} ) eq "ARRAY" ? join ",",@{$\_->{Value}} : $\_->{Value}

}

$prop->{CIMType} type

$prop->{Isarray}  
  
bepaalde property(=attribuut)  
$WbemObject->Properties\_**("attribuutname")->{**Value}  
 ->SystemProperties\_

# Singleton instantie opvragen

Niet met ->get()

$WbemServices->get("$ClassName**=@**") OF ->InstancesOf($ClassName)

# DateTime

* Ge maakt een datetime object aan
* Later vul je de value hiervan in op basis van een attribuut van een bepaalde instantie
* Indien dat bepaald attribuut niet goed geformateerd was moet jet het nog omzetten met regex

use Win32::OLE::Variant;

my $DateTime = Win32::OLE->new("WbemScripting.SWbemDateTime");

$DateTime->{Value} = $Instance->{timeRelatedAttribute};

$DateTime->GetFileTime

#### Omzetten naar juist format

if ($value=~/\\*\\*/){

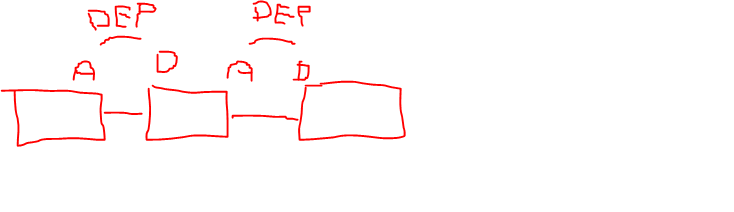
$value=~s/\\*/0/g;

}

$DateTime->{Value} = $value;

# Win32\_DependentService

Voor recursie . antecedent=voorgaande dependent=afhankelijke. Dit is dus de verbindingklasse



Regex => Enkel de naam (alles na = wordt genomen )

my ($Antecedent,$Dependent) = ($Instance->{Antecedent}=~ /="(.\*)"/, $Instance->{Dependent}=~ /="(.\*)"/);

…   
 push @{$AntecedentHash{$Dependent}},$Antecedent;

push @{$DependentHash{$Antecedent}},$Dependent;

…

@Antecedent=@{$AntecedentHash{$Instance->{Name}}};

@Dependent =@{$DependentHash{ $Instance->{Name}}};

print "\t Antecedent Services : " , (join ",",@Antecedent) , "\n";

print "\t Dependent Services : " , (join ",",@Dependent) , "\n";

todo nog goed oefening 33 es checken

# Qualifiers

Hebben Name / Value

## klassequalifier

$Class->{Qualifiers\_}->Item("Provider")

foreach (in $Class->{Qualifiers\_}){}

zien daje al die bitches hebt als ge ne get of execquery doet:

my $Class=$WbemServices->Get($ClassName,**wbemFlagUseAmendedQualifiers**);

of in query vorm

$WbemServices->ExecQuery("select \* from meta\_class","WQL",wbemFlagUseAmendedQualifiers);

bestaat zekere property? Kijken of zowel item() iets geeft als de value ervan

## Attribuutqualifier

foreach my $prop (in $Class->{Properties\_}){  
$prop->{Qualifiers\_}

### valuemaps koppellen hash{valuemap}=actual\_value

if($Qualifiers->{ValueMap}){  
 @hash{ @{ $Qualifiers->Item("ValueMap")->{Value} } } = @{ $Qualifiers->Item("Values")->{Value} };  
}

als( qualifiers een valuemap bevatten ){

@hash {met de @{valuemap array als key}} = @{ array van alle values van die qualifier }

}

# instantie aanmaken

* Ge vraagt aan een klasse de methode\_id via de qualifier die zegt door welke methode hij gemaakt wordt
* Dan vraagje met methods\_ op voor die methode te geven

$methodeCreate = $Class->{Qualifiers\_}->Item("CreateBy")->{Value};  
 $methode = $Class->{Methods\_}->Item($methodeCreate);

Todo oefening 37!!!!!

# Methodes

Methods\_ attribuut voor methode objecten te krijgen

* Name
* Qualifiers\_ bv. Optional, static
* inParameters is SwbemObject (check properties\_)
* outParameters

1. method object maken
2. object maken die invoerparameters worden via properties\_  
    my $MethodInParameters =$Methods->{"Terminate"}->{InParameters};
3. de invoerparameters correct instellen   
    $inParam->{Name} = ...
4. ExecMethod\_( $methodenaam, $inParam) op de klasse doen
5. Output van die methode is object dan in zijn properties\_ de collectie uitvoerparameters heeft  
    my $MethodOutParameters=$\_->ExecMethod\_("Terminate",$MethodInParameters) ;

$ReturnValue=$MethodOutParameters->{ReturnValue};

# Synchrone Eventconsumers

1. SwbemEventSource maken (queue van events) via $wbemservices->ExecNotificationQuery(query)
2. In oneindige lus NextEvent(periode) oproepen (periode=leeg voor oneindig), periode moet veelvoud zijn van WITHIN
3. \_\_Event objecten ofwel \_\_AggregateEvent objecten afhandelen
4. $|=1 onmiddelike output
5. Opletten met warn=3

Skeleton:

$event\_notification = $services->ExecNotificationQuery($query)

While(1){

$event = $event\_notification->NextEvent(5000)

Win32::OLE->LastError() and X or V

} #{targetInstance} {PreviousInstance} 🡨 in geval modification event,, algemeen: event gebruik je dan als instance

# Asynchrone eventconsumers ( voor wanneer je meer dan 1 query nodig hebt)

1. WQL query(ies)
2. SwbemSink COM object(en) initialiseren
3. Gebruikt call mechanisme
4. Koppeling notification query aan SwbemSink object via $wbemservices->ExecNotificationQueryAsync(SwbemSinkObject, WQLQuery)
5. Methodes ter beschikbaar stellen voor SwbemSink object callback

|  |
| --- |
| my $Sink = Win32::OLE->new ('WbemScripting.SWbemSink');  Win32::OLE->WithEvents($Sink,\&EventCallBack);  $WbemServices1->ExecNotificationQueryAsync($Sink, $Query1);  $WbemServices1->ExecNotificationQueryAsync($Sink, $Query2);  my $Console = new Win32::Console(STD\_INPUT\_HANDLE);  $Console->Mode( ENABLE\_PROCESSED\_INPUT);  until ($Console->Input()) { #wachten op input voor te callbacken  Win32::OLE->SpinMessageLoop();  Win32::Sleep(500);  }  $Sink->Cancel();  Win32::OLE->WithEvents($Sink);  sub EventCallBack(){  my ($Source,$EventName,$Event,$Context) = @\_;} |

# Permanente Eventregistratie

* Moet je als admin doen
* Altijd kijken of put\_() gelukt is met lastError (**wbemFlagUseAmendedQualifiers** als parameter)

1. TimerPath spawnen (interval of oneshot) voor periodiek te werken
2. FilterPath spawnen query op onze gemaakte timermachine
3. ConsumerPath spawnen hoe wordt die query afgehandeld?
4. Filter/consumer binding spawnen

Kijken of prop/qual bestaat met ->Item()

*Statitsche klasse*: **aanmaken**:

$instance = $class->SpawnInstance\_();

$instance->{X} = Y;

My $pad = $instance->Put\_();

Print $pad->{path};

**Deleten**:

$WbemServices->Get("$ClassName.$KeyName='$KeyValue'")->Delete();

Powershell

**Nieuw Com object aanmaken ->new()**

$excel=New-Object -ComObject Excel.Application

**Excel instantie een prop opvragen en methode op aanroepen**

$book=$excel.Workbooks.add()

**Select pipes**

select -first 1

select \*

select Name, count

**Where pipes**

where {$\_.status **-eq** "Stopped" **-and** $\_.status -eq “Running”}

-notlike “s\*”

**Group pipes**group Status

**Sort pipe**

sort LastWriteTime -desc

**Bestanden in map onder**

Get-ChildItem

**Expression**

select @{Name="Last Access";

Expression={$\_.LastWriteTime.TimeOfDay}} als ge iets na de “.” Nodig hebt

@{Name="Class";Expression={$\_.SystemProperties\_.Item("\_\_CLASS").Value}}

**Foreach**

foreach{

Write-Host $\_.Name $\_.LastWriteTime.TimeOfDay

}

foreach{

$aantal=$\_.Properties\_.Count

if ($aantal -gt $max){

$max=$aantal

$klasse = $\_.SystemProperties\_.Item("\_\_CLASS").Value;

}

}

# WMI

Initialisatie

$Location=**New-Object** -comobject "WbemScripting.SWbemLocator"  
$service = $Location.**Connect**Server(".","root\cimV2")

Klasse

$klasse=$service.get("Win32\_LogicalDisk")

$klasse.Properties\_

$klasse.SystemProperties\_

$klasse.SystemProperties\_.Item("\_\_CLASS").Value

$klasse.Instances\_()

$alleklassen=$service.execquery("select \* from meta\_class")

$instantie.Properties\_ + $instantie.SystemProperties\_ | foreach{

$\_.Name+" (IsArray="+ $\_.isArray+") "+$\_.CIMType+" - "+ $\_.Value

}

**Qualifiers**

$klasse=$service.get("Win32\_LogicalDisk",**131072**)

$klasse.Qualifiers\_

$klasse.Qualifiers\_.Item("xxx”).Value

Alle qualifiers tonen van alles properties!!

$klasse.Properties\_ | foreach{

$\_.Name

$\_.Qualifiers\_ | foreach{

write-Host " ",$\_.Name, "(",$\_.Value,")"

}

}

**Printen**

Write-host

**Is een qualifier = 1?**

* eerste $\_ is hier een bepaalde property al. Heeft 1 prop abstract? Is die value 1?

If ( ($\_.Qualifier\_ | where { $\_.Name -eq “abstract” }).Value )

**Methods**

$klasse.Methods\_ | select Name

**Overige**

if ( $\_.Qualifiers\_ | where{$\_.Name -eq "Values"} )

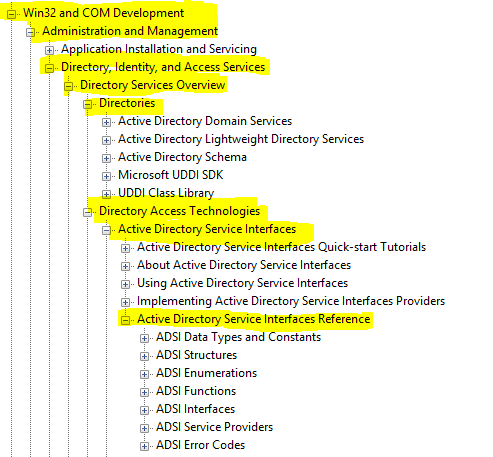
oefening 23!!

$alleklassen=$service.execquery("select \* from meta\_class","WQL",131072)

ADSI

Inloggen hogent ADSI EDIT:

Computer: satan.hogent.be  
advanced: Xandro Vermeulen voor Username/password  
121 1/21



Ldap moniker: zie ADSI LDAP Provider / LDAP AdsPath

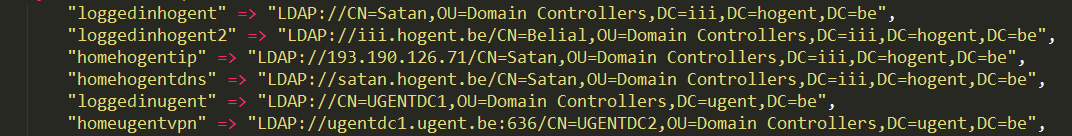
# Stap 0 Bind object methode

* Geeft een ADSI object terug analoog get\_class()
* 2situaties op basis van: uc($ENV{USERDOMAIN} als dat “III” is ben ingelogd op hogent
* Hogent: kijken of het al in moniker vorm is (eerste #chars is dan LDAP://) en daarachter de klassenaam zetten (sub argument) dan via Win32::OLE->GetObject()
* Anders: Via DSO [getobject(“LDAP:”)] ->OpenDSObject() aanroepen.  
  hiervoor zelfde type moniker nodig als hogent met ip er nog tussen + username+pw + ADS\_SECURE\_AUTHENTICATION

# Stap 1 Root object

* Via bind object => RootDSE
* Initialiseren met getInfo()
* Gebruik je voor de domeinnaam op te halen voornamelijk (defaultNamingContext)
* Andere attributen bekijken via MSDN Search: RootDSE (class)

# Stap 2 Connectie met school

* Via BindObject vraag je met de correcte moniker het ldap object

## Klasse opvragen

Bind\_object gebruiken met de full name van de container  
**bind\_object($moniker);**

Belangrijke Attributen (adsi)

|  |  |
| --- | --- |
| * "adspath” (volledig) * "class"(type object) * "GUID" (hash achtig) * “name” | * "name" (voorste v pad) * "parent" (volledig pad verwijend bovenliggend object) * "Schema" |

## Containers in een container opvragen

**$obj** = bind\_object($moniker)  
foreach (in $**obj)**{ print $\_->{**cn**} }

## Filter

$binded\_obj->{**Filter**} = [$classname]

## Get

* ->get() ipv ->{} zodat je foutmeldingen krijgt + getinfo automatisch
* **getEx** indien je array als return waarde wilt (altijd)

# Getinfo

* Getinfo() alles updaten
* getInfoEx([$attribuut1,$attribuut2],0) bepaalde

# subcontainers

subcontainers overlopen van een container via ne filter op organizationalUnit.   
1. Dingen da je wilt updaten met getInfoEx

msDS-Approx-Immed-Subordinate aantal geschatte kinderen ervan

ou naam van de huidige subcontainer

# Property cache

Als alternatief op voorgaande get() shit

Overlopen Item(attribuutnaam/nummer) of next()  
met nummer overloop je via for lus op propertyCount

Wijzigen Put/resetPropertyItem() PurgePropertyList()

Info name / AdDsType / Value (=collectie PropertyValue’s)

PropertyValue get/putObjectProperty(nummertjen)

# Schemaklasse

Bevat attribute of class Schema objecten

Hele schema opvragen via $rootDSE->Get("schemaNamingContext")

Een attribuut of klasse als argument: geeft overeenkomstig schemaobject:   
bind\_object( "cn=" . $argument . "," . $rootDSE->Get("schemaNamingContext") );

* class
* subClassOf afgeleid van
* ObjectClassCatergory ==3 dan hulpklasse
* (system)AuxiliaryClass kan attr overnemen van hulpklasse
* substr( $klasse->{governsID}, 0, 15 ) eq "1.2.840.113556." microsoftspecifiek
* systemOnly kan niet gewijzigd worden
* rdnAttID RDN niet van vorm CN=