

eBdesk

# NEWS Letter

Building Community

September-October 2006



## SPECIAL POINTS OF INTEREST:

- (Web) Services
- WS-I
- WS-I Basic Profile
- WS Platform
- XML, XSD, SOAP
- WSDL
- 3rd-Party
- PHP
- Immune OR Cure?

## INSIDE THIS ISSUE:

- |  |   |
|--|---|
| Web Service Interoperability   | 1 |
| Interoperable WSDL   | 1 |
| 3rd-Party: Always want to be up-to-date with portal's content? Use C-Notifier! | 2 |
| Programming Tips   | 2 |
| Tips & Tricks  | 4 |
| Immune OR Cure?  | 4 |
| eBdesk in a Brief  | 4 |

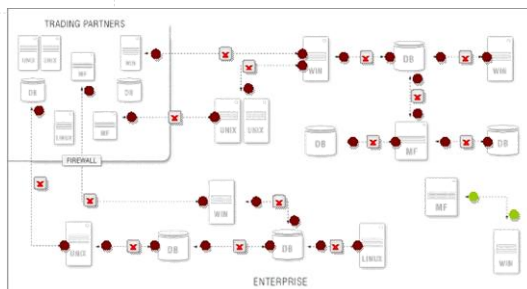
## Web Service Interoperability



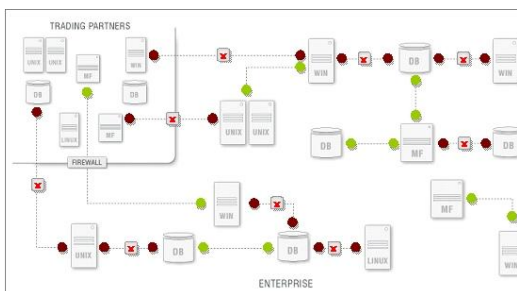
H. Bayu Putera, Software Engineer-ET

One major issue that is always experienced by various business applications is *interoperability* - an ability to operate with other applications. *Interoperability* is required to support the flow of business and to decrease costs by effectively utilizing the available applications. A company that has various business applications, which are probably built by using different *programming languages* and *platforms*, might be experiencing some problems, such as: how to enable those applications to communicate and exchange data. Some efforts to integrate such applications are usually done by handling them case by case basis by implementing *file transfer*, *data parsing*, *message queuing*, or *API*. There is

no global standards that enable two applications to communicate one another with their own different *platforms*, *component models*, and *programming languages*. But then those problems mentioned before can be solved with the development of *Web Service*.



[Illustration] Business applications that are not interoperating one another yet, where most way of communications among systems are blocked. Systems are hard to be interconnected because application points are built separately using different platforms



[Illustration] Business applications that start to connect to Web Service, inside and outside firewall, but still having problem in interoperability as vendors and end-users from each point implement different web services that are likely incompatible to others

The standard of web services allows clients and servers to communicate via HTTP disregard of platforms and programming languages used. Web services guarantee interoperability of applications by implementing a set of open standard specifications i.e.: HTTP, XML, SOAP, WSDL, and UDDI. In reality, the com-

pletion of those specifications left several undefined things or they are only there as option that are not mandatory to implementers. This condition leads to incompatibility of various implementations of web service and soon after generates some issues of interoperability.

Interoperability problems appear in variety and usually consume significant cost and time to be identified.

The failure of web service interoperability in real implementation subsequently become the reason of *Web Services Interoperability Organization (WS-I)* setup, with the aim to solve this class of problem by specifying exactly how HTTP, XML, WSDL, SOAP, and UDDI should be used to ensure

to page. 2 ...

to mention WSDL (*Web Service Description Language*) which is a grammar for explaining web-service's functions and parameters in XML format.

WSDL as one of supporting technologies for interoperability

We will specifically talk about how to create WSDL that support the interoperability of an application.

WSDL (*Web Service Description Language*) is an XML grammar to describe interface of web service, protocols supported by web service, and also their location. WSDL Specification version 1.1 is developed by IBM Research and Microsoft. Although WSDL is not W3C standard or even a mandatory part in developing and invoking web service, but it is widely supported in many SOAP client implementations and development tools.

to page. 3 ...

## Interoperable WSDL



Andi Credo Sibuea, Software Engineer-ET

Web service is an application that is able to expose an API to be accessed over the web. It means that you could invoke the application via web and the application that does the invocation is called the client. It is possible because the standard of web service allows clients and servers to communicate via HTTP although they have different platforms and programming languages. Web Service Platform requires a set of minimum feature specifications to enable to build distributed application. Any platform has its own data representation format and data type system. Web Service Platform must provide a standard data type system to bridge the difference of data type system of various platforms, program-

ming languages, and modeling components.

Web Service Platform must also provide information about interface, function, and parameter descriptions for clients to be used in the calling of the service. Lastly, Web Service must also provide a mechanism to invoke web service remotely. This protocol must be independent of platforms and programming languages.

The technologies used in developing Web Service Platform i.e.: XML (*Extensible Markup Language*) and XSD (*XML Schema Definition Language*) that provide basic format for data representation. Then also SOAP (*Simple Object Access Protocol*) that provide standard mechanism of Remote Procedure Call to invoke Web Services. Not

### 3rd-Party: Always want to be up-to-date with portal's content?

#### Use C-Notifier!

SCAN Tech Lab ([scantechlab@gmail.com](mailto:scantechlab@gmail.com))

eBdesk eXpander portal (Windows), a portal-in-a-box product from eBdesk Corporation (<http://www.ebdesk.com>), is an open-platform portal, containing many ready-to-use applications for (i) messaging, i.e.: mail, calendar, and (ii) collaboration, i.e.: forum, news, image gallery, and many more applications. Unfortunately, eBdesk eXpander Portal did not provide *real-time notification* if there were new contents. Currently, available facility is only channels in Frontpage, which will display newly submitted content(s) for each application. If you are in need of always being up-to-date without always have to refresh the Frontpage, then it is time to install C-Notifier (stand for Comm's Notifier, Comm is the developer's nickname).

#### C-Notifier's Modules and Features

This application consists of two modules, they are:

- **Server Module.** Consists of *notification server* that will serve every request from *Client Module* and some *background applications*, which are functioning as notification data provider. This module is preferred to be installed on the same server where the portal is installed.
- **Client Module.** This *desktop-base* application performs a periodical check on *notification server* to retrieve information whether there is new content or not and display it in a *notification window*, just like Yahoo! Messenger notification when displaying information about a newly online/offline user. This application is installed on portal-user's computer.

This application has the following features:

- Dual checking model. Manual and Automatic.
- Various notification items. Mail, event, task, forum, article, poll, image
- Click & Visit. Enable user to directly login to portal and display the corresponding content (s).
- Sound notification
- Portal Multi-domain support
- Basic Authentication Proxy support

#### Installation Preparation

Installation material can be obtained by sending e-mail to [commedo@gmail.com](mailto:commedo@gmail.com) or [scantechlab@gmail.com](mailto:scantechlab@gmail.com) with the subject: [C-Notifier] Download, and the *e-mail's body* containing information about name, company/organization, company/organization address, active e-mail address (installation material will be sent to this e-mail address), the reason to download this application. If those requirements were not satisfied, then installation material cannot be delivered.

- The following are *prerequisites* for C-Notifier installation:
- Server Module
  - eBdesk eXpander Corporate Portal (Windows)
  - Windows Task Scheduler
  - PHP Module and Command-Line (MSSQL connection support)

- MSSQL
- Client Module
- Windows 98/Me/2000/XP/2003/Vista

#### Installation

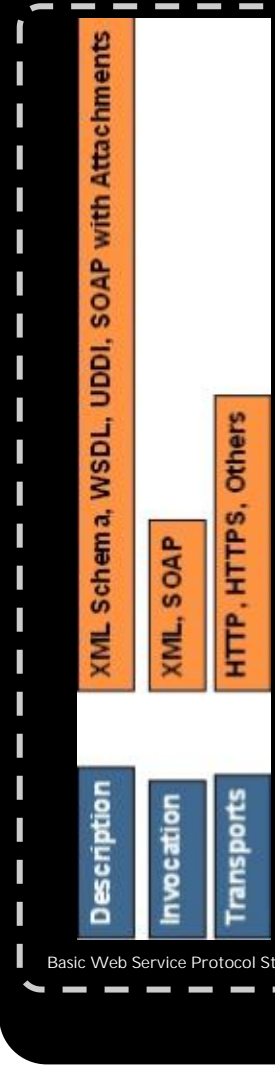
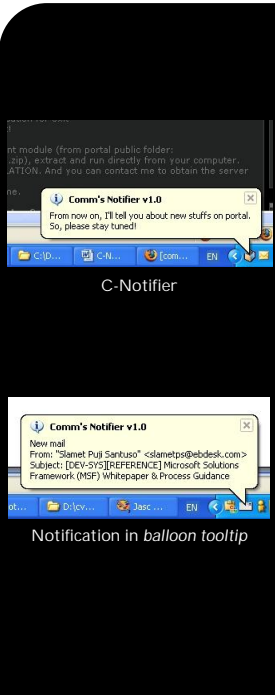
Install *Server Module* on the same server where eBdesk eXpander (Windows) portal is installed. Therefore an *administrator* or other as powerful-role is required to access server and portal installation folder that is commonly located on: C:\Program Files\EBdesk\EBdesk4. Notification server, which is part of this Server Module, is a CGI application made by using PHP (php application). Installation details can be read in install.txt (available in Server Module installation material, C-Notifier-server-v1.0.0.XX.zip, XX = server build number). As to Client Module, it is enough to only extract compressed file (C-Notifier-client-v1.0.0.YY.zip, YY = client build number) and place the extraction result in desired folder.

#### Configuration and Usage

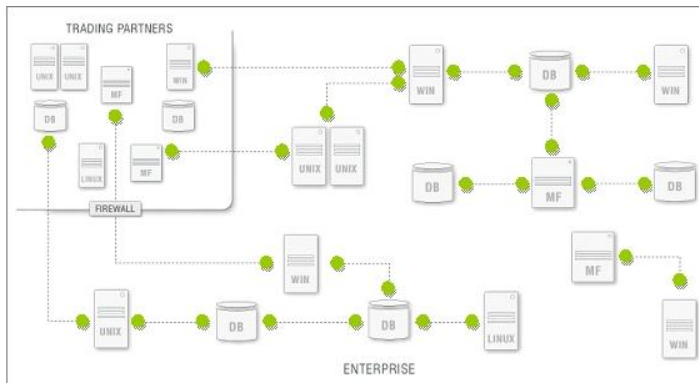
After *server* and *client* installation are finished, it is time to try the application. Before using the application, there are some configurations needed to be done in the first place, i.e.:

- **Checking Items.** By default, will perform checking to all type of available notification.
- **Notification Server URL.** Server address to be contacted by *client*.
- **User Credential.** *User/password* used to access portal.
- **Portal URL.** Although optional, this URL will

to page. 3 ...



## Web Service Interoperability...



[[Illustration] Business applications that start to connect to Web Services that are compliant with WS-I's profile

interoperability between different implementations of web service. This specification is called WS-I Profile. In addition to providing profile, this organization also provides examples of design and application and more importantly WS-I Testing Tools that can be used to verify that our built web service is comply with the WS-I Profile. By referring to WS-I Profile, developer can develop interoperable web service on messaging level and decrease the cost of possible-risks that are likely to appear from the implementation of web service during integration.

design and implementation would greatly support ensuring interoperability and save time and money during application integration.

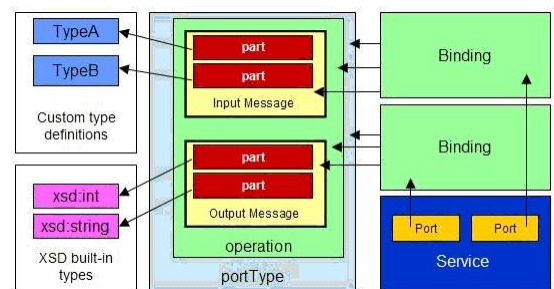
#### Reference

- [1] <http://msdn.microsoft.com/msdnmag/issues/04/11/ServiceStation/default.aspx>
- [2] [http://www.ws-i.org/docs/presentations/wsi\\_cidx\\_preso\\_110105.ppt](http://www.ws-i.org/docs/presentations/wsi_cidx_preso_110105.ppt)

Web service implementers should refer to some specifications in WS-I Profile. Those specifications are divided based on *Web Service Protocol Stack* as shown in the figure beside.

The more items of the WS-I Profile specifications that we fulfill, the higher level of web service interoperability that we obtain.

Interoperability cannot be achieved easily from web service. It requires researches and experiences along with deep understanding about different issues in various web service specifications and various implementation tools. Fortunately, WS-I has done some of the tasks by providing guidance in the form of profile, examples, and testing tools. Using them in



[[Illustration] Content of a WSDL

## Programming Tips

Slamet Puji Santoso, Product VP-ET

- [PHP] Basically PHP does not change variable name, which is passed to a script, except for variable containing period (.), which is automatically converted to *underscore* (\_).
- [PHP] To avoid problems, especially for different PHP default setting, it is better for a developer to use feature which has high backward compatibility such as `$HTTP_GET_VARS` to access HTML form's variable with GET method, although this can be non-activated by setting directive `register_long_arrays` with Off value (PHP 5.0.0).
- [PHP] You can check the value of PHP setting/configuration (php.ini) by using `ini_get(string varname)` function. You can also set the value of a directive by using `ini_set(string varname, string nilaibaru)`.
- [PHP] You can check the existence of a file by using `file_exists(string filename)` function.
- [PHP] PHP provides `file(string filename)` function to read a file easier and store it in an array whose each element hold each line of file (still having a newline character). Or use `file_get_contents(string filename)` if you want to store the content of the read file in a string.
- [PHP] You can also use `fopen()` function to read remote document by using URL (HTTP, FTP) as an input of file name if `allow_url_fopen` directive is On.
- [PHP] You can set the value of `error_reporting` directive at run-time by using `error_reporting(int level)` function.

## Interoperable WSDL...

The added value of WSDL is it is readable by machine and human because it is written in XML. Some development tools can generate WSDL document from your codes and vice versa. If a developer want to create an interoperable WSDL, then the developer need to create a WSDL that is compliant with Basic Profile released by WS-I (*Web Service Interoperability Organization* - <http://www.wsi.org>). Web Services Basic Profile 1.0 represents a publicized description about standards, technologies, and guidance in implementation for some main specifications of web service: XML 1.0, XML Schema 1.0, SOAP 1.1, WSDL 1.1, and UDDI 2.0. This guidance consists of a set of requirements that define how those specifications should be used to develop interoperable web service. The specification document can be obtained from <http://www.wsi.org/Profiles/BasicProfile1.0.html>.

If you want to create WSDL, which is comply with the WS-I Basic Profile, then you could write the WSDL by hand and use the WS-I tools to verify that the WSDL is correct. Writing WSDL by hand can be time-consuming and error prone and you might choose instead to take the auto-generated WSDL provided by your platform. If necessary, the WSDL can be modified to remove any platform idiosyncrasies.

Following is a WS-I Basic Profile-compliant WSDL template that can be used as the basis for your WSDL file. To create a WSDL file from scratch may take longer, but it will ensure that we understand what we are specifying and produce a clean result.

The following WSDL has a single request-response operation and single one-way operation. The example schema defines two types and two corresponding elements that are the "documents" to be carried in the SOAP messages. All items in bold, require to be modified to each of developer's specific situation; XML in normal style can stay as-is in the WSDL.

To convert this WSDL template into your WSDL follow these steps:

- Replace all instances of the string "WebService1" with the name for your web service. For example if your web service is called "Payroll" then "WebService1PortType" would become "PayrollPortType".
- Replace all instances of the URL "http://www.example.com" with your web service's namespace.
- Create XML schema(s) that define(s) global complex type for each XML document that you need to send and receive. You will also need to define a different global element for every operation that contains your XML schema type.

• For each XML document create a "wsdl:message" tag with the "name=" attribute set to "Message YourElementName-Here" and the "element=" attribute to a name referencing the XML schema element from your schema.

• In the "wsdl:portType" tag include operations as required for your web service, be their *request-response* or *one-way*. •Rename the request-response operations after their input and output documents, e.g., "requestResponseOperationInDocOutDoc". For one-way operations rename them "oneWayOperationInDoc".

• Modify the "wsdl:binding" element to contain an operation for each operation defined in the portType above, modifying the soapActions to suit your situation.

• Modify the "wsdl:service" element to include a URL suitable for your platform.

### Interoperable Web Service Validation

Creating some WSDL that is comply with the WS-I Basic Profile, unfortunately, guarantees neither the ability to implement, nor its interoperability. A WSDL document is a specification. Platforms may not support all the features specified in the WSDL, or they may incorrectly implement these features, causing run-time failures.

The following steps should help to check that the WSDL can be implemented and produces web services that can interoperate:

- Use your WSDL documents to create a test web service on your main platform of choice. Place test code in each of the web service's methods such that you ensure full coverage of all elements within the XML documents sent out by the web service.
  - Create a client for your web service. Create test code to exercise all the methods in your web service, with special attention to ensure full coverage of all elements within the XML documents that are sent out. Exercise your test code to verify that your web service is functional on your chosen platform.
  - Now create test clients on your other platforms with which you wish to interoperate. SUGGESTION: Minimally, you should have .NET and a J2EE platform tested. Call the web service from step 1) or the beginning.
  - Now create web services on each of the platforms using the WSDL from the web service created in step 1). Do N by M testing of clients and services.
  - Now publish the WSDL and implement the actual web service code and clients.
- These steps may (and probably will) uncover issues that may cause you to go back and change the WSDL. But they are minimal requirements to validate the developed-WSDL, by referring to WS-I's Basic Profile, to achieve certain level of

```
<?xml version="1.0" encoding="utf-8"?>
<wsi:definitions targetNamespace="http://www.example.com"
xmlns:tns="http://www.example.com"
xmlns:wsi="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
wsi:type="xsd" xmlns:xsd="http://www.w3.org/2001/XMLSchema" >
<xsd:schema elementFormDefault="qualified"
http://www.example.com targetNamespace=" " >
<xsd:element name="aFoo" type="tns:fooType"/>
<xsd:complexType name="fooType">
<xsd:sequence>
<xsd:element name="head" type="xsd:int"/>
<xsd:element name="tail" type="xsd:int"/>
</xsd:sequence>
</xsd:complexType>
<xsd:element name="aBar" type="tns:barType"/>
<xsd:complexType name="barType">
<xsd:sequence>
<xsd:element name="pebbles" type="xsd:int"/>
<xsd:element name="rocks" type="xsd:int"/>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
</wsi:definitions>
<wsi:message name="MessageFoo"/>
<wsi:part name="document" elements="tns:aFoo"/>
</wsi:message>
<wsi:message name="MessageBar"/>
<wsi:part name="document" elements="tns:aBar"/>
</wsi:message>
<wsi:portType name="WebServicePortType">
<wsi:operation name="requestResponseOperationFooBar">
<wsi:input message="tns:MessageFoo"/>
<wsi:output message="tns:MessageBar"/>
</wsi:operation>
<wsi:operation name="oneWayOperationFoo"/>
<wsi:input message="tns:MessageFoo"/>
</wsi:operation>
</wsi:portType>
<wsi:binding name="WebServiceSoap"
type="tns:WebServicePortType"
xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:soapenc="http://schemas.xmlsoap.org/wsdl/soap/encoding/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/" >
<soap:binding transport="http://schemas.xmlsoap.org/soap/http"
style="document"/>
<wsi:operation name="requestResponseOperationFooBar">
<soap:operation
soapaction="http://www.example.com/requestResponseMessage"
style="document"/>
<wsi:input>
<soap-body use="literal"/>
</wsi:input>
<wsi:output>
<soap-body use="literal"/>
</wsi:output>
</wsi:operation>
<wsi:operation name="oneWayOperationFoo">
<soap:operation
soapaction="http://www.example.com/oneWayMessage"
style="document"/>
<wsi:input>
<soap-body use="literal"/>
</wsi:input>
</wsi:operation>
</wsi:binding>
<wsi:service name="WebService" >
<wsi:port name="WebServiceSoap" binding="tns:WebServiceSoap" >
<soap:address
location="http://localhost/WebApplication1/WebService.asmx"/>
</wsi:port>
</wsi:service>
</wsi:definitions>
```

Interoperable WSDL template

interoperability.

### Reference

- [1] *Creating Interoperable Web Service*, <http://devresource.hp.com/drc/resources/creatingInteropWebSvc/>
- [2] *Developer's Guide Revision 1.0 (INA): Web Service Guidelines*, Hasanudin Bayu Putera, eBdesk Technology, 2006

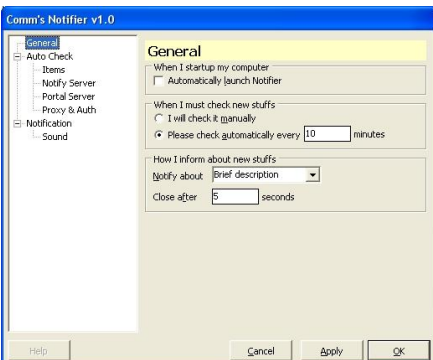
## 3rd-Party: Always want to be up-to-date with portal's content?

### Use C-Notifier! ...

be useful if you want to use *click & view* feature.

Configuration steps as mentioned before:

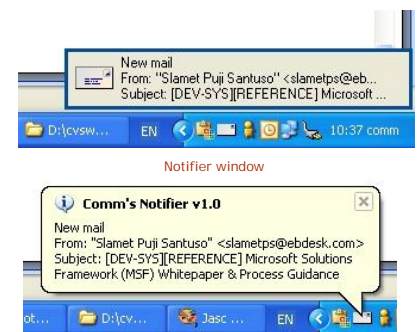
- Checking Items
- Right click on C-Notifier icon on the system tray
- Choose Options menu
- In Options window, choose *Auto Check > Items*. The "All available items" option is checked by default. If you want



Options Window

- only certain item(s), uncheck that option and check only desired item(s).
- Click Apply button to save and apply changes, or click OK to do apply changes and automatically close Options window.
- URL Notification Server & User Credential
- In Options window, choose *Auto Check > Notify Server*.
- Fill in the form on *Server* section:
  - \* Type: HTTP
  - \* Host: expander.ebdesk (host/domain name of portal where notification server is installed)
  - \* Port: 80 (used port number)
  - \* URL: /ebhtml/notify/ (web relative path, if it is not the same path as in the instruction from install.txt, write down other path according to installation)
  - For *User Credential*, fill in *Authentication* section:
    - \* User: portal user id
    - \* Password: password to access portal
    - \* Domain: portal domain name (ask the *Portal Administrator* if you do not know)
  - Portal URL
- In Options window, choose *Auto Check > Portal Server*.
- Fill in the form in *Address* section:
  - \* Host: portal host/domain address
  - \* Port: the default is 80 (if you are not sure, ask your *Portal Administrator*)

Now you can use C-Notifier as shown in the following figures:



Have a nice try!



#### EBDESK INDONESIA ■

Sudirman Tower 19th Floor  
 Jl. Jend. Sudirman Kav. 60  
 Jakarta Indonesia 12190  
 P +62 21 527 0669  
 F +62 21 520 4438

#### EBDESK SOLUTIONS ■

Sudirman Tower 19th Floor  
 Jl. Jend. Sudirman Kav. 60  
 Jakarta Indonesia 12190  
 P +62 21 527 0669  
 F +62 21 520 4438

#### EBDESK TECHNOLOGY ■

Jl. Raden Patah 21  
 Bandung Indonesia 40132  
 P +62 22 250 2423  
 F +62 22 253 3792

#### EBDESK MALAYSIA ■

822, Block A, Damansara  
 Intan 1 Jalan SS 20/27  
 Petaling Jaya 47400  
 Selangor Darul Ehsan,  
 Malaysia

#### Website

<http://www.ebdesk.com>

#### Email

[marketing@ebdesk.com](mailto:marketing@ebdesk.com)  
[partner@ebdesk.com](mailto:partner@ebdesk.com)

#### Worldwide Office / Partners

Malaysia, Singapore, Thailand,  
 Philippines, Japan, Saudi  
 Arabia, Germany, USA, UK,  
 Canada, Australia.

eBdesk is established in 1998, in the beginning of Internet business. Set out with corporate portal product development, nowadays eBdesk grows into a business enabler company with its various products, solutions, and services. The fall down of global business and the collapse of dotcom business in its first year of establishment was indeed a knock out and difficult time, but not only did eBdesk survived, it also arose and grew more mature over and over again.

eBdesk organization consists of eBdesk Ltd as holding company, separated into Product & Technology, International Operation, and Solution & Services. The organization reflects our vision, mission, and strategy to develop products, deliver solutions and services for international market. On May 2005, the number of clients has reached more than 100 all over the world, mostly are big companies having more than 10.000 employees.

eBdesk cooperates with partners in marketing, implementation, and solution development area, as part of international partnership. This involves distribution agreement, product bundle, and joint venture to set up a new company.

There are four major eBdesk products:

- eBdesk eXpander Corporate Portal  
 This application package consist of two primary components:  
 - Platform. The basic foundation for add-on application to run.  
 - Collaboration. Add-on application for collaboration such as mail, calendar, forum, news, etc.
- eBdesk eXpedition Workflow  
 A tool which user can design, simulate, implement, supervise, and manage workflow of many distinct business processes.
- eBdesk eNcyclo Document Management  
 Organizes all types of documents, from its creation, approval, documentation, and distribution.
- eBdesk eXpert Knowledge Management  
 Manages data, information, and knowledge cycle. Has taxonomy, spider, expert, and community development.



## TIPS & TRICKS

Vera Firmansyah, Senior Engineer-ET

### eBdesk Products

- Logo. To change logo is quite easy by accessing *Personalization > Display > Emblem* page. But, how if the logo is becoming annoying to you while you are interacting with eBdesk Portal? You may simply change the logo with a (*transparent*) blank image.
- Custom Channel (R2). If you are an admin, how to activate *custom channel* in Frontpage? Please access *Administration > Access-Right > Report and Graph* page.
- Channel/Tab Title. You want to change the word "Frontpage", "Mail", etc with some quite easy enough remembered words? How to do that? Go to *Personalization > Page > Content* page. Type those words in *Custom Name* and do not forget to press *Apply* button.
- Mail Signature (R2). eBdesk Webmail Client provides 10 *customizable signatures*. You can set your signatures in *MyDesk > Mail > Settings*. In *Personalization* option, choose *Signature*.

### Popular Application

(Mozilla) Firefox

A *multi-platform web browser* which is more popular and appear more mature. It convincingly challenges MSIE domination.

- [CTRL]+[T] to open new tab.
- [CTRL]+[F4] atau [CTRL]+[W] to close active tab.
- [CTRL]+[TAB] to move between tabs from left to right.
- [CTRL]+[SHIFT]+[TAB] to move between tabs from right to left.
- [CTRL]+[U] top display HTML source code of the active document.

(Mozilla) Thunderbird

One of *multi-platform desktop mail client application* as one of descendants of Mozilla project with high-level of flexibility (*extension*).

- [F5] to retrieve new *e-mail* of an *account*.
- [CTRL]+[SHIFT]+[T] to retrieve new *e-mail* of all *accounts*.
- [CTRL]+[N] atau [CTRL]+[M] to compose new *e-mail*.
- [CTRL]+[E] atau [CTRL]+[R] to *reply* selected *e-mail*.
- [CTRL]+[L] to *forward* selected *e-mail*.

## Immune OR Cure?

Slamet Puji Santoso, Product VP-ET

Which Anti-Virus do you need?

If you are an administrator responsible for servers and workstations or even network, Anti-Virus is one of mandatory software that you need to install, especially if you are using MS Windows operating system. There are some things that you need to know before deciding which Anti-Virus that you should use.

There are some common methods used by Anti-Virus to detect the existence of virus, such as:  
 • Heuristic algorithm. Based on virus general behavior. Have potential ability to detect virus, which is not formerly detected by Anti-Virus company.

• Signature Definition List. The most commonly used method, but have drawback that is only able to work out on previously known virus. Depend greatly on periodical update/patch of anti-virus data including for its own security hole.

For main consideration, choose Anti-Virus which provides immediate periodical update of virus data, especially the one using Heuristic Algorithm method!

### eBdesk Products Updates

- eBdesk eXpander Corporate Portal
- R3 Mail (3rd-party & integration)
- R3 Calendar (Service & Web Client)
- eBdesk eXpedition Workflow
- R2 Pack #3: Load-on-Demand, Mass Processing, Visual Flow, Multi-language

[www.ebdesk.com](http://www.ebdesk.com)

eBdesk Technology Jl. Raden Patah No. 21 Bandung Indonesia 40132 Tlp. 022-2502423 Fax. 022-2533792