

Inside the Individual Information Worker Model

This bulletin is broken down to following content:

1. [Overview](#)
2. [IIW model vs. Consumer model](#)
3. [How to support IIW model](#)
 - a. [IIW tenant customer types and service plan](#)
 - b. [Simplified model based on the current business model](#)
4. [Inside the IIW model](#)
 - a. [Active Directory services update](#)
 - i. [IIW Reseller Org creation](#)
 - ii. [IIW Tenant Org creation](#)
 - iii. [Other helper features](#)
 - b. [Email services update](#)
 - i. [SMTP domain related update](#)
 - ii. [Enable mail service for a tenant organization](#)
 - c. [Unauthenticated sign up for IIW](#)
 - d. [Support Office Communication Service, Windows SharePoint Services and other future features](#)
5. [Terminology](#)
6. [Credits](#)

Overview

HMC 4.0 and previous solutions have shipped with two different and segmented user models, commonly known as the Business Model and Consumer Model. These models applied to the way how organizations and users were configured and granted permissions, how resources such as stores were allocated and the features available to users in each model. But the consumer model was a very limited model with respect to the feature set and the ability to up-sell users to better or more diverse services. Consumer users want richer experience, limited collaboration between friends and family, to take advantage of additional services such as Windows SharePoint Services, Office Communication Services. To give consumers or newly called “individual information workers” the same up-sell capabilities that Business users have today, IIW model was introduced in HMC 4.5.

IIW model vs. Consumer model

The following table summarizes the difference between IIW model and Consumer model:

Area	IIW model	Consumer model
AD hierarchy	Hosting OU Reseller A Business Tenant1 User1 User2 ... UserN	Hosting OU Reseller A Business Tenant1 User1 User2 ... UserN

	<p>Business Tenant2</p> <p>User1</p> <p>User2</p> <p>...</p> <p>UserN</p> <p>IIW Reseller C</p> <p>IIW Tenant1</p> <p>User 1</p> <p>User2</p> <p>...</p> <p>UserN</p>	<p>Business Tenant2</p> <p>User1</p> <p>User2</p> <p>...</p> <p>UserN</p> <p>Consumer OU</p> <p>User1</p> <p>User2</p> <p>...</p> <p>UserN</p>
Email Service	<ul style="list-style-type: none"> • Users can log on to Outlook Web Access. • Users can log on to POP or IMAP clients. • Users have access to other users within the same tenant organization public details, for example, Phone Number and Email Address. • Users can log on to MAPI clients including Microsoft Outlook. • Users can look up and view other users within the same organization, in the Outlook Global Address List or using Outlook Web Access Address List feature, In this case of a Consumer or Individual user may include friends and family or external contacts. 	<ul style="list-style-type: none"> • Users can log on to Outlook Web Access. • Users can log on to POP or IMAP clients.
SharePoint Service	Take advantage of hosted services such as SharePoint.	Do not support
OCS Service	Take advantage of hosted services such as Office Communication.	Do not support

Table 1 - IIW model vs. Consumer model

How to support IIW model

IIW tenant customer types and service plan

To support IIW model, we added two customer types "IIWResellerOrg" and "IIWTenantOrg", and also one customer service plan "IIWPlan". You can see them in the following figures:

Table - dbo.CustomerTypes					
	CustomerType...	CustomerTypeName	CustomerTypeDescription	DateCreated	DateLastUpdated
▶	BU	BusinessUser	Customer is a user in a business organization	2/18/2008 4:05:10 AM	2/18/2008 4:05:10 AM
	BZ	BusinessOrganization	Customer is a business organization	2/18/2008 4:05:11 AM	2/18/2008 4:05:11 AM
	CO	ConsumerOrganization	Customer is a consumer organization	2/18/2008 4:05:17 AM	2/18/2008 4:05:17 AM
	CT	Contact	Customer is a contact	2/18/2008 4:05:22 AM	2/18/2008 4:05:22 AM
	CU	ConsumerUser	Customer is a consumer user	2/18/2008 4:05:19 AM	2/18/2008 4:05:19 AM
	GP	Group	Customer is a group	2/18/2008 4:05:20 AM	2/18/2008 4:05:20 AM
	IO	IIWTenantOrg	Customer is an IIW Tenant organization	2/18/2008 4:05:13 AM	2/18/2008 4:05:13 AM
	IR	IIWResellerOrg	Customer is an IIW reseller organization	2/18/2008 4:05:16 AM	2/18/2008 4:05:16 AM
	RO	ResellerOrganization	Customer is a reseller organization	2/18/2008 4:05:14 AM	2/18/2008 4:05:14 AM
	UM	UnifiedMessagingCust	Unified Messaging Customer	2/18/2008 4:25:14 AM	2/18/2008 4:25:14 AM
	UU	UnifiedMessagingUser	Unified Messaging Customer User	2/18/2008 4:25:14 AM	2/18/2008 4:25:14 AM
*	NULL	NULL	NULL	NULL	NULL

Figure 1 - Two new IIW customer types in PlanManager dbo.CustomerTypes table

	PlanID	PlanTypeCode	PlanName	PlanDescription	StatusTypeCode	DateCreat
1	FA60882B-F652-4B61-B3FF-17D28EF05433	AC	ConsumerPlan	An Active Directory Consumer Org	EN	2008-02-1
2	467C91E3-8D6E-420D-9B5F-8C486FB9343A	AB	BusinessPlan	An Active Directory Business Org	EN	2008-02-1
3	87B2228F-CEFF-4B98-A9FD-BA98347A3D3F	AB	IIWPlan	An Active Directory IIW Tenant Org	EN	2008-02-1

Figure 2 - A new IIW customer service plan in PlanManager dbo.ServicePlans table

Simplified model based on the current business model

With IIW model added for provisioning, in Managed Active Directory::GetPolicy procedure, we remove the legacy MultiGroup container whose contents provide a smaller AD footprint for representing both our hosted businesses model and consumer model. The MultiGroup container is only for some legacy support in Windows 2000 or previous Operating Systems. It is actually no longer existed in Windows 2003 and beyond. By removing this container, all new organizations are created without the overhead of this additional OU structure per organization. At the same time, we will be deprecating our existing consumer model, leaving the API support in place but removing any reference to this model from our Docs.

Previous AD hierarchy for a Business organization:

```

OU=<TenantOrg>
  OU=_Private
    CN=MultiGroup
      CN=UserN
      CN=AllUsers@<TenantOrg>
      CN=ChildOrgN
    CN=Services
    CN=AllUsers@<TenantOrg>
  CN=Admins@<TenantOrg>
  Users
  Custom or service specific groups

```

Current AD hierarchy for Tenant Model:

```

OU=<TenantOrg>
  OU=_Private
    CN=Services
    CN=AllUsers@<TenantOrg>
  CN=Admins@<TenantOrg>
  Users
  Custom or service specific groups

```

Inside the IIW model

Active Directory services update

IIW Reseller Org creation

CreateIIWResellerOrganization

If you want to experience IIW features, here is the starting point. We provide a new procedure Hosted Active Directory::CreateIIWResellerOrganization. It is actually a wrapper of procedure Managed Active Directory::CreateOrganization. In addition, the CreateIIWResellerOrganization sets the organization's AD property businessCategory to "IIWReseller" through which it could be distinguished from standard Reseller. It also sets customerTypeCode in DB to the "IR", which stands for "IIWResellerOrg". The policy applied to the IIW reseller is the same as a standard reseller.

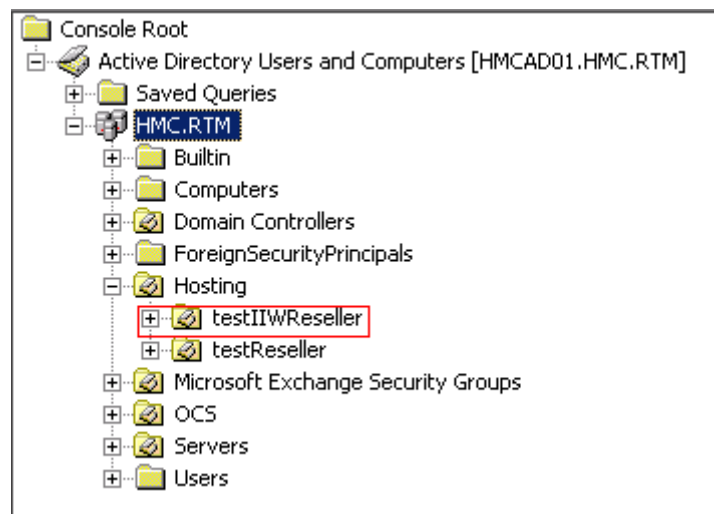


Figure 3 - An IIW reseller named "testIIWReseller"

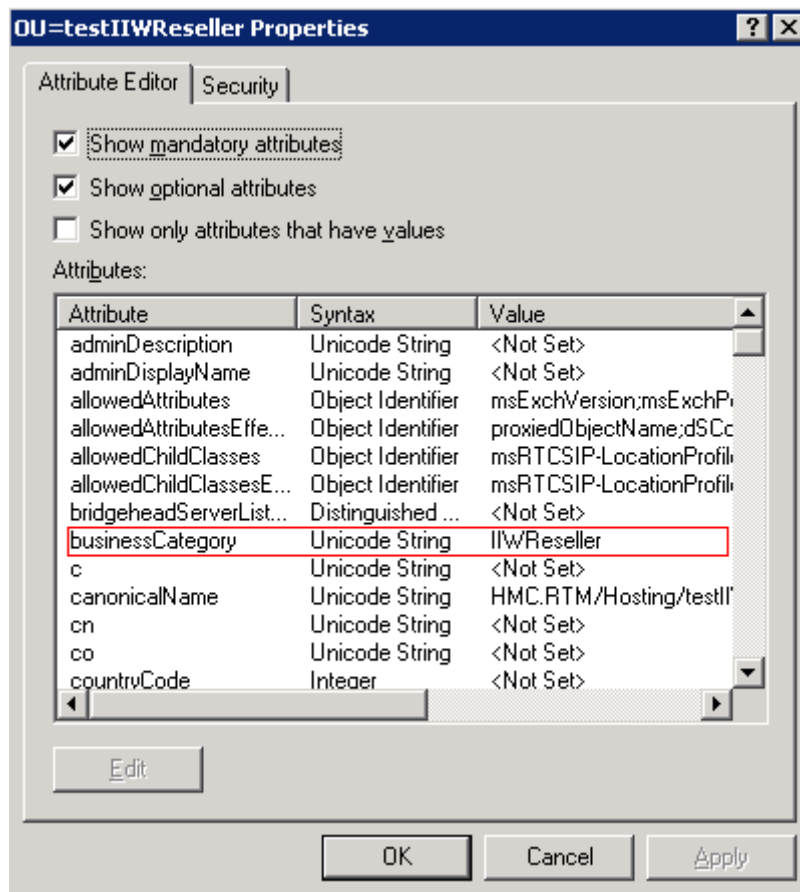


Figure 4 - IIW reseller's AD property businessCategory set to "IIWReseller"

IIW Tenant Org creation

CreateIIWTenant

Once an IIW reseller is created, IIW tenant organization can be created under that reseller. We provide a new procedure Hosted Active Directory::CreateIIWTenant. It also wraps the procedure Managed Active Directory::CreateOrganization. In addition, CreateIIWTenant sets the organization's AD property businessCategory to "IIWTenant", sets customerTypeCode in DB to "IO" which stands for "IIWTenantOrg", subscribes the organization with "IIWPlan" customer service plan. Notice that CreateIIWTenant creates not only a tenant organization but also an organization admin.

An IIWTenant organization and its admin are created as shown in the following figure:

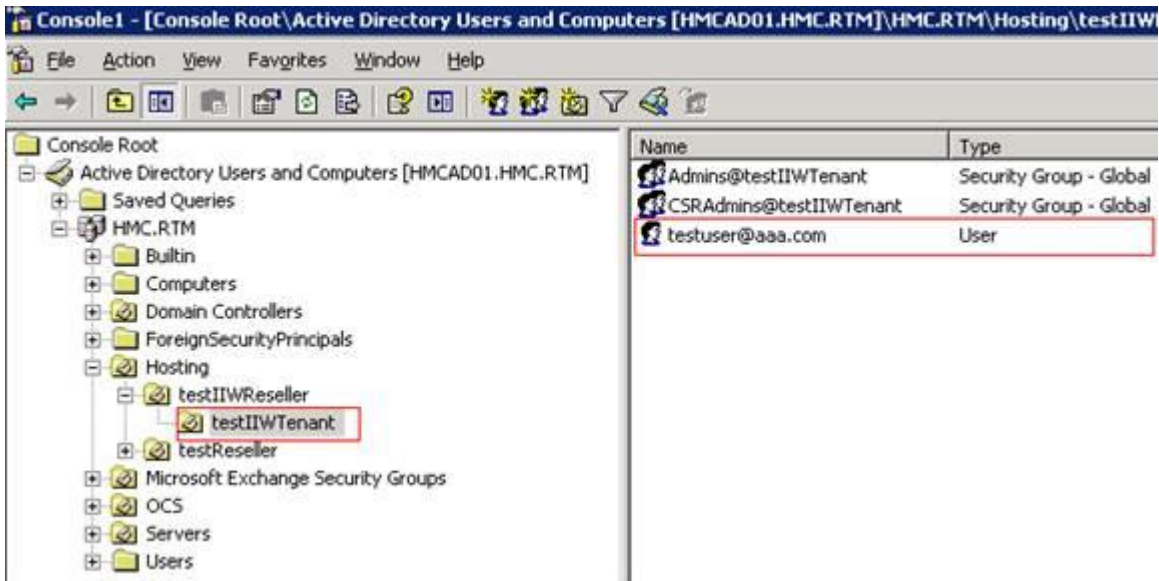


Figure 5 - An IIW tenant named "IIWTenant" and its admin "testuser@aaa.com"

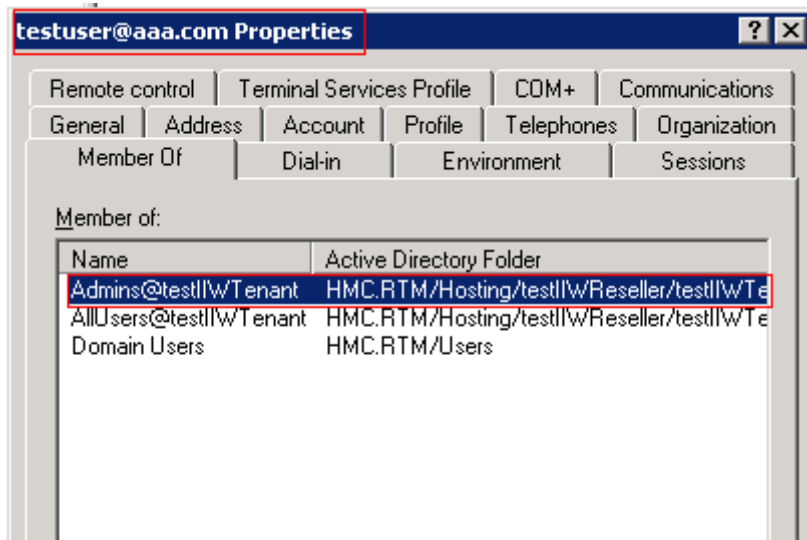


Figure 6 - The "testuser@aaa.com" user is the tenant org admin

Here is an IIW tenant organization's AD property page:

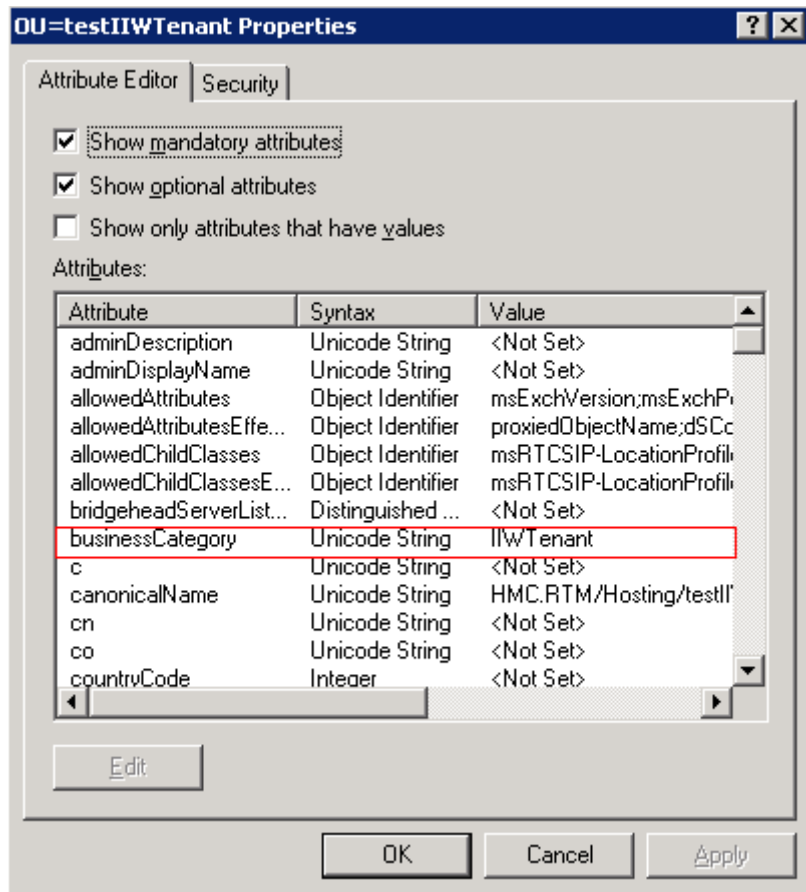


Figure 7 - IIWTenant's AD property businessCategory set to "IIWTenant"

After an IIW reseller and an IIW tenant are created, three customer records will be created in PlanManager dbo.Customers table:

	CustomerTypeCode	CommonName
1	BU	testuser@aaa.com
2	IO	testIIWTenant
3	IR	testIIWReseller

Figure 8 - An IIW reseller, IIW tenant and admin user in Customers table

Other helper features

GetTenantRoot

This procedure returns the domain name of the tenant organization object when you supply an AD object's LDAP path. It is just a wrapper of Managed Active Directory::GetThisOrganizationRoot.

1. If the LDAP path is a business user, returned root is the DN of the business organization.
2. If the LDAP path is an IIW tenant user, returned root is the DN of the IIW Tenant OU.
3. If LDAP path is an organization, the DN itself is returned.

This procedure mainly allows for easy abstraction of the Tenant OU from the user. This enables the ability that UI or higher level namespace can easily identify and act against the IIW tenant organization when supplied only the path of an IIW user object.

UserIsSelf

If this procedure is supplied a user object LDAP path or UPN, it checks whether the user who submits the request is the supplied user object (LDAP path or UPN which you provided in request). This procedure is intended to support a common requirement of control panels and self service portals. Here are some user scenarios that may need this procedure:

1. Display a change password UI

2. Display a reset password UI
3. Check whether allow an admin user to delete himself or not

In detail, this procedure actually checks if the submitted SID is the same as the user sending this request. Submitted SID is set to securityContext/@trustee by default. But if you provide authentication/basic node in request, the submitted SID will be overridden by value authentication/basic/@username.

A sample request with authentication/basic node and trustee attribute is listed below:

```
<request>
  <data>
    <user>LDAP://CN=Administrator,CN=Users,DC=hmc,DC=rtm</user>
    <preferredDomainController>HMCAD01.hmc.rtm</preferredDomainController>
  </data>
  <procedure>
    <execute namespace="Hosted Active Directory" procedure="UserIsSelf" impersonate="1">
      <before source="data" destination="executeData" mode="merge" />
      <after source="executeData" destination="data" mode="insert" />
    </execute>
  </procedure>
  <context>
    <securityContext trustee="hmc\Administrator">
      <authentication>
        <basic username="hmc\Administrator" password="Pass1word" />
      </authentication>
    </securityContext>
  </context>
</request>
```

Email services update

SMTP domain related update

The following three SMTP domain related public procedures are changed to support rollup of SMTP Domain ownership. If a reseller organization owns an SMTP domain, all Tenants of that reseller can also own that SMTP domain as an asset in the Managed Plans database. In previous version, you could only create SMTP domain either on reseller level or tenant level. With these changes, you could not only share a SMTP domain from Reseller but also create a vanity domain for your tenant organization.

CreateSMTPDomain

This Hosted Email namespace procedure was modified to handle the scenario while creating SMTP domain for Tenant organization where the specified domainName has already been created as an accepted domain and is owned by the parent reseller organization. Here we provide a scenario matrix to make the behavior more clear:

Scenario	Result
Domain does not exist in any organization. Create for Reseller/IIWReseller.	Reseller/IIWReseller is the owner. CustomerAsset is added.
Domain does not exist in any organization. Create for BusinessOrg/IIWTenantOrg.	BusinessOrg/IIWTenantOrg is the owner. CustomerAsset is added.
Domain exists and owner is Reseller/IIWReseller. Create same domain for the Reseller/IIWReseller.	Owner is unchanged. Bypass the created CustomerAsset.
Domain exists and owner is Reseller/IIWReseller. Create same domain for a child BusinessOrg/IIWTenant.	Owner is unchanged. CustomerAsset is added for the child

	organization.
Domain exists and owner is BusinessOrg/IIWTenant. Create same domain for the BusinessOrg/IIWTenant.	Owner is unchanged. Bypass the created CustomerAsset.
Domain exists and owner is BusinessOrg/IIWTenant. Create same domain for parent Reseller/IIWReseller.	Owner is unchanged. An error message is thrown from GetSMTPDomain.
Domain exists and owner is Reseller/IIWReseller. Create a vanity domain for child BusinessOrg/IIWTenant.	Reseller level domain owner is unchanged. Vanity domain's owner is the BusinessOrg/IIWTenant.

Table 2 - CreateSMTPDomain scenario matrix

The minimal role to use this procedure is UserCreators.

DeleteSMTPDomain

This Hosted Email namespace procedure was modified according to the CreateSMTPDomain change.

A scenario matrix is shown in the following table:

Scenario	Result
Domain exists and owner is Reseller/IIWReseller. Delete same domain from the Reseller/IIWReseller.	Domain does not exist. CustomerAsset removed.
Domain exists and owner is Reseller/IIWReseller. BusinessOrg/IIWTenant has this domain as customer asset. Delete same domain from a child BusinessOrg/IIWTenant.	Owner is unchanged. Only BusinessOrg/IIWTenant's CustomerAsset was removed.
Domain exists and owner is BusinessOrg/IIWTenant. Delete same domain from the BusinessOrg/IIWTenant.	Domain does not exist. CustomerAsset removed.
Domain exists and owner is BusinessOrg/IIWTenant. Delete same domain for parent Reseller/IIWReseller.	Owner is unchanged. An error message is thrown from GetSMTPDomain.

Table 3 - DeleteSMTPDomain scenario matrix

The minimal role to use this procedure is UserCreators.

GetSMTPDomain

This procedure is actually a wrapper of a private procedure named ValidatedOrgOwnsDomain_ (where the real code logic change happens). The procedure is modified to return an additional <ownership> node that indicates whether the SMTP Domain is owned at the tenant or reseller levels.

A scenario matrix is shown in the following table:

Scenario	Result
Domain exists and owner is Reseller/IIWReseller. Get SMTP domain with Reseller/IIWReseller as the org.	Ownership is Reseller/IIWReseller
Domain exists and owner is Reseller/IIWReseller. Get SMTP domain with child BusinessOrg/IIWTenant as the org.	Ownership is BusinessOrg/IIWTenant
Domain exists and owner is BusinessOrg/IIWTenant. Get SMTP domain with BusinessOrg/IIWTenant as the	Ownership is BusinessOrg/IIWTenant

org.	
Domain exists and owner is BusinessOrg/IIWTenant. Get SMTP domain with parent Reseller/IIWReseller as the org.	An error message such as “The organization does not own the SMTP Domain with the name {SMTPDomainName}” is thrown

Table 4 - GetSMTPDomain scenario matrix

The minimal role to use this procedure is UserCreators.

Enable mail service for a tenant organization

MailEnableTenant

This procedure helps to enable a tenant organization for Hosted Email Services. In this procedure, it batches three operations including subscribe Hosted Email Service, add available plans and create SMTP domain. In fact, this procedure is not only used for IIW organization but also business organizations. However, it does have some special handling abilities for the IIW organization, specifically the ability to identify parent organization from a user path (GetTenantRoot is called if provided a user path instead of an organization path in MailEnableTenant request), in order to subscribe the appropriate object. In summary, the procedure is just a wrapper of the following procedures:

- Hosted Email 2007::Subscribe
- Hosted Email 2007::AddAvailablePlans
- Hosted Email 2007::CreateSMTPDomain

If you do not call this procedure, you need to call those three procedures separately in your code. Inside this procedure, it calls Hosted Active Directory::GetTenantRoot to get the owning organization LDAP path for mail service enable and then subscribe with the organization plan name, add available user plans and create a SMTP domain for the tenant organization. Then the tenant organization can use mail service, you can create the mailbox for the users under the tenant organization using the added available user plan. The tenant organization’s users can also use OWA features.

Unauthenticated sign up for IIW

IIWSignup

This procedure is modeled very closely to a legacy procedure BusinessSignup. IIWSignup creates an IIW Tenant Organization and Admin user as well as optionally generates a password and sends an email to a user alternate email address with the password. There is no explicit role check in this procedure. Actions are performed under the context of a privileged account. The account is already created and configured as an executeAS credential during Hosted Signup::Initialize. This procedure could be exposed through a Web Service that is configured to allow Anonymous access. Customers are strongly encouraged to protect calls to this procedure using secured web services (locked down to only receive requests from a single source for example) as well as anti-scripting measures such as credit card validation implemented at the UI layer. As indicated above this feature is in place to support the development of Web Portals that allow for anonymous or unauthenticated sign up. There is a great deal of responsibility placed on the developer of the Web Portal to further secure this sign up process beyond the suggestions presented above.

Support Office Communication Service, Windows SharePoint Services and other future features

IIW model interacts with the HMC environment in much the same way as business model does. One primary difference may be the active hours, with IIW users interacting with the system more after typical business hours. User can take advantage of any service on the platform including Office Communication Service, Windows SharePoint Services and even other future features. This may be the most important reason why we use IIW.

Terminology

AD	Active Directory
HMC	Hosted Messaging and Collaboration
IIW	Individual Information Worker
IMAP	Internet Message Access Protocol
MAPI	Messaging Application Programming Interface
OU	Organization Unit

Credits

The bulletin is contributed by:

Author: Yunpeng Song, Linden Goffar

Technical Reviewer: Yonghong Shi, Jinliang OU, Steve Li

UA Reviewer: Tony Liu