The Grid-in-a-Box services are meant to provide basic remote execution capabilities for the WSRF.NET toolkit. Grid-in-a-Box consists of 5 services and 2 client applications. Two of the services, the ExecService and the DataService, should be placed on every machine in your grid. The three other services, the AccountService, the ResourceAllocationService and the ReservationService have only 1 instance per Virtual Organization in your grid. The two client applications are the GridAdminClient and the GridClient. The GridAdminClient is used by a VO’s administrator to specify which applications are available to run on the VO’s resources and which users are VO members. The GridClient is used by VO users to reserve VO resources, stage data to/from remote hosts and start jobs on those remote hosts.

The general architecture of Grid-in-a-box is shown in Figure 1.

**NOTE:**
Grid-in-a-box is still a work in progress, as is this document. Although the Grid-in-a-box services provide sophisticated functionality for remote job execution, they are not as robust as the underlying WSRF.NET toolkit and they provide a demonstration of capabilities, not a production grid system. All the source is included, so have fun! Certain capabilities are missing, such as a deployment system. By default, all Grid-in-a-box services are installed on every machine that installs WSRF.NET. In practice, there would be many more installations of the ExecService andDataService than the others.
Grid-in-a-Box Setup

Setting up Grid-in-a-Box requires deploying the ExecService and DataService to each machine in the grid and the ResourceAllocationService, ReservationService and AccountService to a single machine that will “run” the VO. Each instance of the ExecService and DataService, as well as the single instances of the ResourceAllocationService and AccountService, must be configured.

Configuring the DataService:

1. Create a physical directory that the DataService will use to store its data
2. Give the account that your web service runs as (for Windows XP, this is typically ASPNET) permission to write to this directory (and all sub-directories)
3. Edit the DataService’s web.config file (in {WSRFdotNet}\Src\Services\Grid\DataService where {WSRFdotNet} is the directory where you installed WSRF.NET)
   a. Modify the basePath parameter to point to the directory you created in step 1.

Configuring the ExecService:

4. Create an account under which jobs on the ExecService’s machine will run (Start -> Control Panels -> User Accounts). You will typically want to set this account type to “limited”, but depending on the applications you wish to run and your security setup, you may choose the “computer administrator” account type. If you choose “limited”, the account will be limited in where it can read/write and which applications it can execute.
5. Give this account read/write permission to the directory you created in step 1 (and all sub-directories)
   a. Right click on the folder in Windows Explorer, then select “Properties”
   b. Click on the “Security” tab (if you do not see a “Security” tab, then from Windows Explorer choose Tools -> Folder Options, click on the “View” tab and uncheck the box labeled “Use Simple File Sharing (Recommended)”)  
   c. Click “Add”  
   d. The “Select the object type” box should be set to “Users, Groups or Built-in Security Principals”. The “From this location” box should be set to the name of your local computer.
   e. Enter the name of the account in the box labeled “Enter the object names to select”.
   f. Click the “Check Names” button and you should see the account name become underlined. If the account name cannot be found, an error dialog will appear.
   g. Click “OK”.
   h. Make sure the “Read”, “Write” and “List Folder Contents” boxes are checked.
   i. Click “OK”.
6. Add permission to for this account to execute every application that you want grid users to be able to launch on the machine.
   a. For each application, right click on the .exe and select Properties.
   b. Click the “Security” tab.
   c. Click “Add”.


7. Edit the ExecService’s web.config (in {WSRFdotNet}\Src\Services\Grid\ExecService where {WSRFdotNet} is the directory where you installed WSRF.NET):
   a. Modify the username and password service parameters to correspond to the account you created in step 3.
   b. Modify the DataService, ResourceAllocationService, ReservationService and AccountService parameters to contain the appropriate URLs for those services in your deployment. The default value for the DataService parameter points to the DataService on the local machine. The default values for the ResourceAllocationService, ReservationService and AccountService parameters are macros which also point to the local machine. The DataService value will likely not have to be changed because an ExecService and a DataService should be deployed on each machine in the VO. The other three parameter values must be changed to point to the machine that is hosting those three services for the VO.
   c. Modify the <subscription-manager-url> parameter to point to the correct SubscriptionManager service (note: the default value is a macro which points to the SubscriptionManager service on the current machine and will not need to be changed for a typical WSRF.NET install).

Configuring the AccountService:
8. Modify the AccountService’s policyCache.config file (in {WSRFdotNet}\Src\Services\Grid\AccountService where {WSRFdotNet} is the directory where you installed WSRF.NET). Note that unlike the ExecService and DataService, there is only one AccountService for the VO, so you only need to perform this configuration on one machine.
   a. Toward the bottom of this file find the series of XML elements:
      <wssp:Integrity>
      <wssp:TokenInfo>
      <wsp:OneOrMore>
      Inside the <wsp:OneOrMore> element, you will see multiple <wssp:SecurityToken> elements. These will be modified in steps b and c.
   b. Modify the value of the <wssp:SubjectName MatchType="wssp:Exact"> element (under the <wssp:Claims> element) to contain the DN of the certificate of the Administrator of the VO.
   c. Modify the value of the <wssp:X509Extension OID="2.5.29.14" MatchType="wssp:Exact"> element to contain the key identifier of the certificate of the Administrator of the VO. Note that the values for both of these elements can be found using the WSE 2.0 X.509 Certificate Tool as shown in Figure 2. Also, the certificate(s) of the administrator (or administrators) must be stored under “Local Computer” in the Microsoft Certificate Store (as shown in Figure 2).
d. Note: the default policyCache.config use the <OneOrMore> element to show how to specify multiple administrators. Follow steps 8b and 8c for each administrator’s certificate. If you do not have multiple administrators, remove the second <wssp:SecurityToken> element.

Configuring the ResourceAllocationService:

9. Modify the ResourceAllocationService’s policyCache.config (in \{WSRFdotNet\}\{WSRF\} \{WSRFNet\}\Grid\ResourceAllocationService where \{WSRFNet\} is the directory where you installed WSRF.NET).
   a. Modify the <wssp:SecurityToken> elements as discussed above in steps 8a - 8d.

Administering Grid-in-a-Box

Grid-in-a-Box is administered using the GridAdminClient. This client allows the administrator to set up VO membership and application information. This client is shown in Figure 3 and the binary is located in WSRF.Net’s bin directory (i.e. the bin directory where WSRF.NET was installed). To use this client, first put the URLs for the ResourceAllocationService and the AccountService that are used by this VO in the type-in boxes at the top of the GUI.
Figure 3. The GridAdminClient

Updating VO Membership:
The GridAdminClient can be used to both add and remove members in the VO. Members are allowed to run applications (execute jobs) and store data on remote machines. When a vendor is added to the VO, they are allowed to run on all of a VO’s ExecServices.

To add a new user:
1. Put the DN for the new user in the “Enter new user’s DN” box on in the “VO Membership” section of the GUI. This can be done in two different ways.
   a. Type in the DN as it appears in the “Subject” box on the WSE X.509 Certificate Tools (shown in Figure 2).
   b. If the certificate for the new user is installed in the “Current User” section of the Microsoft Certificate Store on the machine on which the GridAdminClient is running, click the “Browse for certificates” button and select the certificate for the user you wish to add to the VO. The certificate’s DN will appear in the “Enter new user’s DN” box.
2. Click the “Add VO member” button
   a. If this is the first action you have taken since you started the GridAdminClient, you will be prompted to select a certificate for the VO administrator (i.e. the certificate with the DN you put in the policyCache.config files of the AccountService and ResourceAllocationService). The selected certificate will be used to sign the outgoing message to the AccountService. Using the wrong certificate will cause the AccountService to reject the message.
b. When the AccountService has correctly added the new user, you should see “VO member added” in the “Status” box

To remove a VO member:
1. Either type the DN into the “Enter new user’s DN” box or browse to that certificate using the “Browse for certificates” button.
2. Click the “Remove VO member” button.
   a. You will see “VO member removed” in the “Status” box when this operation completes.

**Updating VO Applications:**
The VO administrator must tell the ResourceAllocationService what applications are deployed in this VO, and where. An application is a Windows binary. The application can be deployed on 1 or more of the hosts in the VO. Each application will have an application name that is exposed to the VO members, as well as an application path that represents the location of the binary for that application. If the same binary is deployed on multiple hosts in the VO, each must be added through the GridAdminClient and it is expected that the same application name will be used for each.

To add an application:
1. Enter a name for the application in the “Enter Application Name” box in the “VO Applications” section of the GUI. This name will be the name that end users will use to refer to the application. Note that if the same application is deployed on multiple machines in the VO, the same name should be entered in this box for each ExecService that can run the application.
2. Enter the path to the application binary on the remote machine in the “Enter application path” box. This path should be an absolute path for the file system accessible by the machine on which the application is deployed. If the GridAdminClient is running on a machine that shares a file system with the machine on which the application is deployed, the “Browse” button can be used to find the binary and fill in this box.
3. Enter the URL of the ExecService running on the machine on which the application is deployed in the “Enter ExecService name” box. Note, do not use “localhost” in the URL - you must use the actual machine name.
4. Enter the URL of the DataService that is running on the same machine as the ExecService entered above, in the “Enter DataService name” box. Note: do not use “localhost” in the URL - you must use the actual machine name.
5. Click the “Add Application” button
   a. You should see “Application added” in the “Status” box if everything completes successfully.
6. Repeat these steps for every machine (i.e. every ExecService which can execute the binary). Then repeat for every application available to VO members.

To remove an application:
1. Enter the application name in the “Enter Application Name” box.
2. Enter the ExecService in the “Enter ExecService Name” box.
3. Click the “Remove Application” button
   a. You should see “Application removed” in the “Status” box.

Using Grid-in-a-Box

The GridClient allows VO users to reserve execution services (ExecService’s), stage data to/from remote machines and launch computational jobs. The GridClient GUI is divided into two sections, as shown in Figure 4.

![Image of GridClient GUI](image)

**Figure 4. The GridClient**

The Reservation section allows users to gain exclusive use of ExecServices (which are running on various machines in the VO). The user will reserve a service (or services) based on the application they want to run. The ResourceAllocationService will generate a list of all ExecServices that can run the user’s desired application and which are not already reserved. From this list, the user can choose which ExecServices they would like to reserve, and request that the ResourceAllocationService reserve them on their behalf. In order to help the user choose which ExecServices to reserve, they can query the ExecService for a standard set of data about the host on which it is running. The user can also retrieve their current reservations to see which ExecServices they have already reserved. It is important to note the relationship between the application name used to find an ExecService and a reservation of an ExecService. A reservation means that only the reserving user can run jobs using the reserved ExecService (i.e. only the user can run on the machine hosting the ExecService). However, the reservation is not specific to the application. Nothing prevents a reserving user from running a different application on a reserved machine (assuming the application they actually run is installed on the reserved machine) and no other user is allowed to run any application on that machine.
Making Reservations:
1. Put the URL of your VO’s ResourceAllocationService in the type-in box under “Reservation”.
2. Enter the name of the application you wish to run in the box labeled “Application Name”. This name should correspond to the name used by the Administrator when adding/updating VO applications.
3. Click the “Find Available Resources” button.
   a. The names of available ExecServices that can run the desired application will be displayed in the “Available Resources” box.
4. Place check marks in the boxes next to the ExecService you wish to reserve.
   a. If you want to find out more information about the machine on which an ExecService is running, highlight that ExecService (in the Available Resources box) and click the “Resource Information” button. Information will appear in the Messages box as shown in Figure 5.
5. Click the “Make Reservation” button.
   a. You will be asked to select a certificate (if you have not click on another button that already required you to select your certificate) to use to sign the message to the ResourceAllocationService. This certificate must be one whose DN has been added to the AccountService using the GridAdminClient.
   b. When your reservation is complete, an icon will appear in the box labeled “Your Current Reservations”. Clicking on this icon will display information about your reservation (see Figure 6).

![Figure 5. Reserving Resources using the GridClient](image)
Discovering Reservations:

1. If you want to discover the reservations that you have already made (for example, because you have just started the GridClient), click the “Get Current Reservations” button.
   a. You may be prompted to select your certificate (if you have not already done so since you started with GUI).
   b. The reservations will be displayed as icons in the “Your Current Reservations” box, as shown in Figure 6.

Once a reservation is made, the user will want to upload data and run jobs on the reserved ExecService. Each ExecService has an associated DataService to which the input data for the job should be uploaded. In order to do this, the user must first select the reservation to use.

Selecting a reservation & ExecService to use:

1. Highlight one of the reservation icons in the “Your Current Reservations” box.
   a. The title of the box displaying the reserved ExecServices changes to say “Select a reserved service to use”.

2. Highlight one of the ExecServices listed.
   a. Any actions taken in the “ExecService” portion of the GUI will be relative to this ExecService. Any actions taken in the “DataService” portion of the GUI will be relative to the DataService associated with the selected ExecService.

Creating a DataService:
1. Select an ExecService as discussed above (this will automatically target the corresponding data service)
2. Click the “Create New Data Resource” button.
   a. A new text icon will appear in the “My Data Resources” box.

Uploading files to a data resource:
1. Highlight a data resource in the “My Data Resources” box.
2. Click the “Upload File” button.
3. You will be prompted with a file dialog box and you can browse to the file you wish to upload.
   a. “File Uploaded” will appear in the “Files” box when the upload is complete.

Displaying the files in a data resource:
1. Highlight a data resource in the “My Data Resources” box.
2. Click the “Get File List” button.
3. The “Files” box will show the files that are part of the data resource with their size in bytes displayed in parenthesis. This is shown in Figure 7.

![Figure 7. Using a Data Resource](image)

Retrieving files from a data resource:
1. In order to retrieve files from a remote data resource to your local file system, first display the files in that data resource as shown above.
2. Highlight the file you wish to transfer in the “Files” box.
3. Click the “Retrieve File” button.
a. You will be prompted with a file dialog box. Browse to the location on your local file system where you would like to save the file, enter the local filename and click OK.

Deleting a data resource:
1. To remove a data resource (and all files stored within it), highlight the data resource in the “My Data Resources” box.
2. Click the “Delete Data Resource” button.

Discovering data resources:
1. Highlight a reserved ExecService.
2. Click the “Get My Data Resources” button.
   a. Each data resource will appear as an icon in the “My Data Resources” box.

Once ExecServices have been reserved, a data service has been created and files have been staged in, a remote job can be launched. This is done using the ExecService portion of the GUI.

Launching a job:
1. Highlight an ExecService.
2. Highlight a data resource.
3. Type the application name in the “Application Name” box.
4. Type the job arguments in the “Job Arguments” box.
5. Click the “Start New Job” button.
   a. The binary associated with the Application Name you typed will be started with its working directory set as the directory corresponding to the data resource you highlighted. The job’s arguments will be the Job Arguments given.
   b. An icon representing this job will appear in the “My Job Resources” box.
   c. When the job is done, an asynchronous notification will be sent to the GridClient. This notification will contain the job’s exit code, exit time and run time and will be displayed in the “Job Information” box. This is shown in Figure 8.
   d. The ExecService will also send a message to the ReservationService to remove the ExecService from your reservation (though other ExecServices may still remain).
   e. You can now retrieve output files from the job’s data resource using the “Retrieve File” button as discussed above.
Polling for Job Status:
1. A job resource’s status can be determined by selecting the job icon in the “My Job Resources” box.
2. Click the “Job Status” box.
   a. The status will be shown in the “Job Information” box. Note that as long as the job resource exists, the job status will be retrievable – even if the job has exited.

Discovering Job Resources:
1. Current job resources owned by a particular user can be discovered (such as when the GridClient has just been started) by clicking the “Get My Job Resources” button.
   a. You will be prompted to select your X509 certificate.
   b. An icon for each job resource will appear in the “My Job Resources” box.

Deleting Job Resources:
1. Select a job resource icon in the “My Job Resources” box.
2. Click the “Delete Job Resource” button.

Figure 8. The GridClient after a Job Completes