Texas Guaranteed

To be the premier source of information, financing, and assistance to help all families and students realize their educational and career dreams.

AdvanTG Web v1.0

Application Build and Deploy Process

AdvanTG Web v1.0 Software Support

Release Date: 09/04/2002

This document represents the procedures necessary to monitor, maintain, upgrade, or perform system-wide administration for the AdvanTG Web v1.0 system. Texas Guaranteed Student Loan Corporation classifies this document as *Confidential and TG Proprietary Information*.

Copyrights and Confidentiality

Trademarks

AdvanTG Web, the AdvanTG Web logo, AdvanTG, Loans by Web, and the TG logo are trademarks of Texas Guaranteed Student Loan Corporation.

Sun, Sun Microsystems, the Sun logo, iPlanet, the iPlanet logo, Java, JAR, and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

IBM, the IBM logo, Tivoli, the Tivoli logo, DB2, NetView, Tivoli Enterprise, and Tivoli Enterprise Console are trademarks or registered trademarks of International Business Machines Corporation or Tivoli Systems Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, Microsoft Internet Explorer and the Windows logo are trademarks of Microsoft Corporation in the United States or other countries.

Netscape and Netscape Navigator are registered trademarks of Netscape Communications Corporation in the United States or other countries.

BEA and WebLogic are registered trademarks of BEA Systems, Inc. BEA WebLogic Server is a trademark of BEA Systems, Inc.

NetIQ, and the NetIQ logo are trademarks of NetIQ Corporation in the United States.

F5 and BIG-IP are registered trademarks of F5 Networks, Inc.

All other company and product names mentioned are used only for identification purposes and may be trademarks or registered trademarks of their respective owners.

Document Revisions

AdvanTG Web v1.0 Documentation Changes

Revision	Date	Author	Revisions Made
0.6	07/08/02	Smith, Bowen, Blackard, et. Al and others	Initial Draft
0.7	07/11/02	Eric Schneider Robert Blackard	Add procedure for Staging Deployment Add procedures for Pre-Prod
0.8	07/18/02	Lance Smith	Add reference sections & retrieval of archived build
0.9	08/07/02	Lance Smith	Incorporate Technical Reviews Blackard, Bowen, Roush, and Awan
1.0	08/15/02	Lance Smith	Add information for preserving B2B storage folders when new **Staging** AdvanTG Web builds are done
1.1	08/29/02 09/04/02	Lance Smith	Add services table and `log4j-config.xml` info Verify services shut-down and start-up sequence Add list of AdvanTG software services on page 13.

Document Identification

File Name   E:\2002_Texas_Guaranteed\AdvanTG Web Build and Deploy User Guide.doc

VSS Location $AdvWeb/MasterBuild/AdvanTG Web Build and Deploy User Guide.doc

Last Changed   9/4/2002 10:57 AM

Changed By   Smith, Bowen, Blackard, et. Al

Security   Confidential and TG Proprietary Information

Status   Revised -

Document Version   v1.1

Table of Contents

Figures and Tables

Overview and Background Information

This document covers the detailed procedures for building and deploying AdvanTG Web application components to the development, staging, and pre-production servers.

Intended Audience

Texas Guaranteed Student Loan Corporation personnel responsible for:

- Integrating emergency software fixes into the AdvanTG Web II business solution
- Incorporating new server configurations, software upgrades, or hardware changes
- Updating the AdvanTG Web II software with requested enhancements

Terminology

Ant
An open-source Java build tool (without the headaches of "make") that uses XML syntax files. This means that anyone conversant with basic XML structures and formatting can comprehend and write an Ant file. (See *Related Visual SourceSafe Documents and Internet Sites*)

Artifact
An artifact is anything that is a deliverable of the development process. This includes the use case documents, the analysis model, the design model, source code, configuration files, etc. It also includes any deliverable built from other artifacts, for example, WAR files, JAR files, EJB-JAR files. *Source* artifacts are files not built from some other artifact.

Build Manager
Person who starts the AdvanTG Web software build, verifies its success, notifies developers of any errors, resolves build problems, labels successful builds, and maintains build scripts.

Code Manager
One of several persons is responsible for a portion of the AdvanTG Web source code in the product. Prior to the code freeze the code manager has overall responsibility for the quality of their assigned code and assists other developers working on the code. After the code freeze, the code manager is the only person who can check out assigned code from Visual SourceSafe. Once the code manager has checked out a source file, they can direct another developer to make file changes. However, the code manager must review any changes prior to checking the code back into Visual SourceSafe. The code manager is also responsible for identifying what re-testing must be performed in response to a code change.

Code Freeze
The date on which general development for a product release concludes. After the code freeze only critical defects and change requests approved by the Release Manager will be implemented. A code freeze may be a partial freeze, applying only to specified source code modules, or a total freeze, applying to all of the source code.

Configuration Item
A configuration item is one file that is under version control in Visual SourceSafe. Configuration items are grouped into folders in Visual SouceSafe. Folders can contain other folders. The leaf nodes are the configuration items.

Dependent Item
Any part of the overall configuration of an execution environment that is not maintained in Visual SourceSafe. The execution environment depends on the item, but it is not retrieved from Visual SourceSafe. Examples include the operating system, software packages required to execute AdvanTG Web, DB2 database, and mainframe accounts. These are not stored in Visual SourceSafe, yet are required for proper AdvanTG Web operation.

EJB – Enterprise JavaBeans
An architecture for setting up program components, written in the Java programming language, that run in the server parts of a computer network that uses the client/server model. Enterprise JavaBeans are built on the JavaBeans technology for distributing program components to clients in a network. Enterprise JavaBeans offers enterprises the advantage of being able to control change at the server rather than having to update each individual computer with a client whenever a new program component is changed or added. EJB components are reusable in multiple applications.

Execution Environment
The environment contains all of the hardware, software and configurations required to execute the application. This includes the software developed as part of the project as well as all dependent item software (that is , WebLogic Server, Java Virtual Machine, etc.)

Folder
A folder is a Visual SourceSafe file structure that can contain configuration items, and possibly other folders. Same as a file directory (Windows) or Project (Visual SourceSafe).

JAR – Java ARchive
A file that contains the class, image, and sound files for a Java application gathered into a single file and compressed (zip file format) for faster downloading or processing. When a programmer gets a Java program development kit, a small program or utility called "jar" is included. The jar utility lets you create, list, and extract the individual files from a JAR file.

Label
A name assigned to a collection of configuration items. AdvanTG Web uses the standard format `TEST_mmyyddvv`, where `mm` = month, **dd** = day, `yy` = year, and **vv** = desired build version (usually **01**, **02**, etc.) The same collection can later be retrieved using this label. (See *AdvanTG Web Build Label Standard* on Page 8.)

Release
A collection of configuration items in Visual SourceSafe, each marked with a specific version number, and the artifacts built from them. A release is labeled so that it can later be recreated. During development, the release does not identify all the dependent items (and their versions) required.

Release Manager
Person responsible for managing an AdvanTG Web product release. The release manager works with project management and business sponsors to determine what business functionality will be in a release. The release manager sets the date of the code freeze. After the code freeze the release manager works with project management and business sponsors to ensure that only critical defects and change requests are implemented. The release manager coordinates the implementation of critical defects and change requests with the code managers.

VSS – Visual SourceSafe® Repository

The Microsoft Visual SourceSafe software that serves as storage container for all AdvanTG Web II source artifacts. It contains all versions of these artifacts. Users can retrieve a specific artifact version or insert new versions of an artifact. It prevents multiple concurrent check-outs of the same component by two project members simultaneously.

WAR – Web ARchive

A file that contains the HTML, images, and Java Server Pages files for a web application gathered into a single file and compressed (zip file format) for easier deployment.

Related Visual SourceSafe Documents and Internet Sites

- *Building & Deploying AdvanTG Web Project*
  `$/AdvWeb/System Documentation/System Administration Guide/`
  `AdvanTG Web Server Administration Quick Reference.doc`

- Key Ant Script that controls the building processes
  `$/AdvWeb/MasterBuild/build.xml`

- AdvanTG Web Global Build Parameters
  `$/AdvWeb/MasterBuild/buildinclude.properties`

- `Http://jakarta.apache.org/ant`
  Information concerning Ant XML-based scripts

Security Required to Deploy AdvanTG Web

In order to install this software, you need to have Windows **Administrator** rights on the servers within the proper target environments:

Table 1: Environment Servers

Environment	Servers Used
Falcon Development	Falcon
Staging	ADVS-IIS1, ADVS-BEA1, ADVS-BEA2, ADVS-IPRPT
Pre-production	Hydra1 & Hydra2, Libra1 & Libra2, Areies1 & Aries2, Virgo1 &Virgo2, Moray, DB2/UDB

You can check to see if you have administrator rights on your machine by running the **User Manager** application which can be located from the Start menu:
`Right-click My Computer and select Manage`→ `Local Users and Groups.`

Double-click on your name and click the **Groups** button to see if you are a member of **Administrators**. If you do not have administrator rights, contact x4999 and ask them to come back and give you administrator rights on your machine.

AdvanTG Web Build Label Standard

All AdvanTG Web builds are labeled with a 13 character name that follows this standard:
TEST_mmyyddvv,
where **TEST_** is mandatory, fixed portion of the label,
mm = current month number,
dd = day number of the current month,

yy = current year, and
vv = build version for the day (**01**, **02**, etc.)

Main Steps in Build/Deployment Operations

Procedure: General Steps

1) Stop all affected services on the target machines. (See *Reference: Services on AdvanTG Web Servers* on page 33) for a quick reference to the names of AdvanTG Web services running on the network servers.

2) Extract the latest copies of script files from Visual SourceSafe:
   ```
   $/AdvWeb/MasterBuild
   $/AdvWeb/dev/AdvWeb/build6.1     (6.1 denotes the WebLogic version)
   $/AdvWeb/dev/Services/build6.1
   $/Security/Phase 2/TGSLC/build6.1
   ```

3) Determine if there have been any changes within the `AdvWeb.properties` file for the target machine. This can often be a source of problems!

4) Run the build and deployment scripts.

5) Restart the necessary services within the target environment servers.

Figure 1: Staging and Pre-production Environment
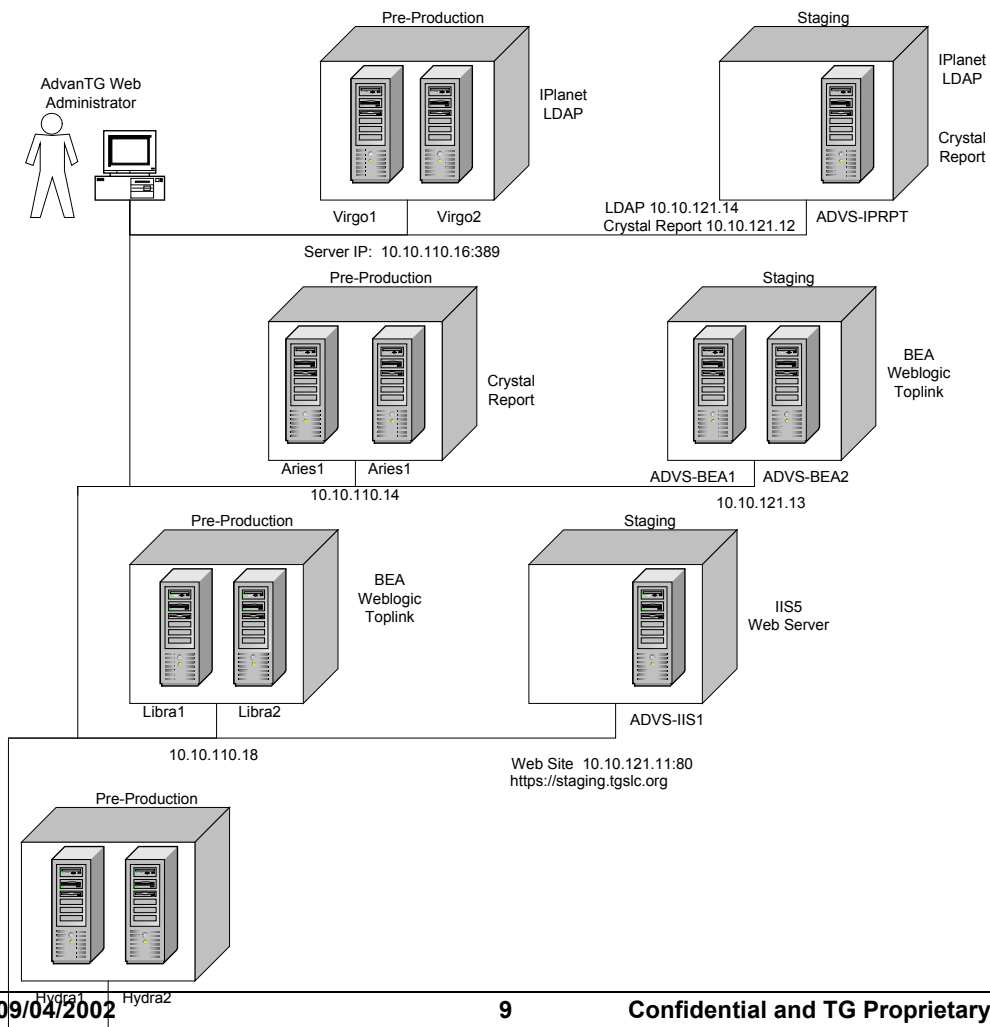
Build Scripts

> **Note:** The "**6.1**" designation in the Visual SourceSafe folders refers to version 6.1 of the WebLogic server.

Extract recursively from Visual SourceSafe to your local drive, these file directories:

- `$/AdvWeb/MasterBuild`
- `$/AdvWeb/Development/AdvWeb/build6.1`
- `$/AdvWeb/Development/Services/build6.1`
- `$/Security/Phase 2/TGSLC/build6.1`

The build process was written using Ant, an open-source Java build tool from the Jakarta project. The Ant Jars (executable class files) are found in the $/`AdvWeb/MasterBuild/ant` folder of Visual SourceSafe.

> **Caution:** At this time, the build process does not automatically extract Build Scripts from Visual Source Safe, this is dependent on human intervention.

Visual SourceSafe Setup

Initialization File

For the Visual SourceSafe commands in the Ant scripts, you need to ensure that the initialization file contains the proper statements. If the `srcsafe.ini file` on your machine does not contain the correct location of the Visual SourceSafe server, the Ant script cannot automatically extract source from the repository. The file name is:

```
D:\Program Files\Microsoft Visual Studio\VSS\srcsafe.ini
```

It *must contain* the include statement:

```
#include \\Marlin\SrcSafe\Ads_Dev\SRCSAFE.INI
```

Administrative Security

When the Ant script extracts the source and artifacts from Visual SourceSafe, it needs to use a Visual SourceSafe **userID** and its corresponding **password**. This account must have adequate file permissions to read the files from all Visual SourceSafe directories involved in the build process. This can be changed by editing the file named -

```
$/AdvWeb/MasterBuild/buildinclude.properties
```

Change these lines in this file

vssafe.userid=xxxx

vssafe.password=xxxx

> **Note:** See Mitchell Reid (extension 4657) for the required Visual SourceSafe UserID necessary to perform builds and deployments.

Mapping Local Machine Directories to Visual SourceSafe Folders

File directories on the local PC workstation *must be mapped* to corresponding Visual SourceSafe folders for the build and deployment scripts to work.

Manual Creation of File Directories on Workstation

- Create a directory named `D:\AdvWeb`
- Create a directory named `D:\AdvWeb\dev`
- Create a directory named `D:\Security`

Manual Creation of Working Folders in Visual Source Safe

In Visual SourceSafe (VSS), a working folder for a project directory can be set by first selecting the project folder in the directory tree, then select **Set Working Folder** command from the **File** Menu.

- Set the working folder for `$/AdvWeb` → `D:\AdvWeb`
- Set the working folder for `$/AdvWeb/Development` → `D:\AdvWeb\dev`
- Set the working folder for `$/Security` -→ `D:\Security`

Figure 2: Visual SourceSafe Root Source Folders

AdvanTG Web   **Single Sign-On Web Portal**   **MasterBuild**
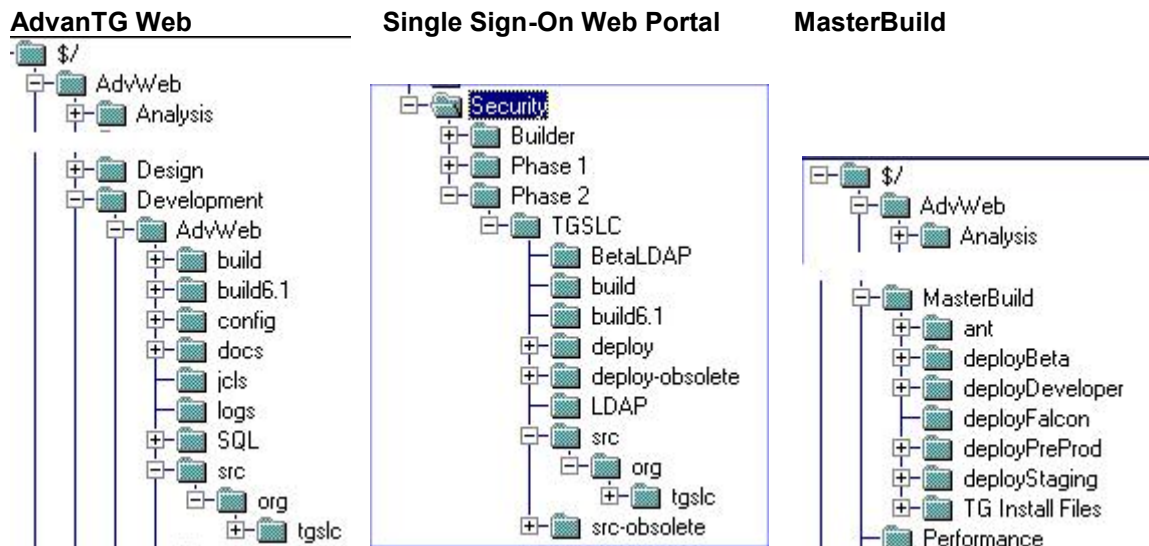
Batch Files Used for Building AdvanTG Web

- There are six types of batch files used in the AdvanTG Web build process which results in the compiled artifacts being stored in - `\AdvWeb\MasterBuild\staging`
- `build.cmd` - compiles, creates Jars, Wars, Zip files with the source files stored locally on the hard drive. It retrieves from Visual SourceSafe any sources that are not stored locally. An example of this behavior, is the third party libraries which need to be in the directory: `d:\AdvWeb\MasterBuild\staging\thirdPartyLib`

- **createJavaDocs** - creates the JavaDocs HTML files in the local directories ..\docs\api relative to the four build directories:

  ```
  \AdvWeb\dev\AdvWeb\docs\api
  \AdvWeb\dev\Deploy\docs\api
  \AdvWeb\dev\Services\docs\api
  \Security\Phase 2\tgslc\docs\api
  ```

- **extractAndBuildAll.cmd** - *deletes* source on your local drive and extracts from Visual SourceSafe the latest version, compiles, creates Jars, Wars, and Zip files. *[Run this Batch file from MasterBuild root directory only!]*

  Caution: This batch file will delete out any code you are currently modifying. Be careful. One recommendation is to rename your active development source directories prior to running this batch command.

- **label.cmd** - Labels all source code. *[Run only by the Release Manager!!!!]*
  Example command usage: **label TEST_12130101**
  The format of the labeling is **TEST_MMDDYYNN**, where **NN** is the number (01, 02, etc.) of the build for that day. (See *AdvanTG Web Build Label Standard* on Page 8.)

- **extractLabel.cmd** - Extracts a specific label and builds all the required artifacts. This function is only available at the MasterBuild level. Currently the build scripts are not extracted based on a label - it assumes that the build scripts on the local machine are the *correct scripts* to build a label. The build scripts are labeled when running the **label.cmd** and can be manually extracted to build a label with the build scripts of that label. *[Run this Batch file from MasterBuild root directory only!]*

Example command usage: extractlabel TEST_12130101

extractLabelAndArchive.cmd – Performs the functions of **extractLabel.cmd** and places the artifacts in a directory containing a historic archive of build artifacts. *[Run this Batch file from MasterBuild root directory only!]*

General Deployment Notes

Environment Services You Must Know

During the AdvanTG Web application deployment, you will stop and start specific services on several of the servers. This table defines the server names and corresponding AdvanTG Web services used in the TG AdvanTG Web environments. For detailed information about controlling the software services such as iPlant Directory Server, B2B Server, WebLogic Server, the IIS web server, and special logging services (Steelhead, Chinook), please consult the *AdvanTG Web Server Administration Quick Reference.doc*.

Note: Services that are indented under the column **AdvanTG Web Services**, are *dependent* (child) services that require the *parent* service to start, prior to running the *child* service.

Table 2: Which Servers Run Specific AdvanTG Web Services

AdvanTG Web Services	Staging	Beta	Production
RMI Service B2B Service	Falcon	ADVS-BEA1 ADVS-BEA2	Libra1 & Libra2
RMI Service Beta B2B Service Beta	Falcon		
Chinook Syslog	Falcon		

IIS Admin Service
  FTP Publishing
  Simple Mail Transport
       Protocol (SMTP)
  World Wide Web Publishing Service
	Falcon
	Falcon
	Falcon
	
	Falcon
		ADVS-IIS1
		ADVS-IIS1
		ADVS-IIS1
		
		ADVS-IIS1
			Hydra1
			Hydra1 and Hydra2
			Hydra1 and Hydra2
			
			Hydra1 and Hydra2

IPlanet Administration Server	Falcon	ADVS-IPRPT	Aries1
IPlanet Directory Server 5	Falcon	ADVS-IPRPT	Aries1 & Aries2
AdvanTG Web Domain Services myserver , guiserver AdvanTG AdvBeta	Falcon	ADVS-BEA1 ADVS-BEA2	Libra1 & Libra2
Parallel Crystal		ADVS-IPRPT	Virgo1 & Virgo2
Steelhead	Falcon	All Servers	All Servers

What Happens During Deployment

While specific actions happen within the different AdvanTG Web environments, this table provides an overview of what the deployment scripts are accomplishing within each of the major development, staging, and pre-production environments

Table 3: Deployment Operations by Environment

Description of Deployment Operations	Development Mydomain advantg	Falcon	Staging	Pre-Prod
Copy the AdvWeb War	No[1] Yes	Yes	Yes	Yes
Copy other War files (security)	Yes	Yes	Yes	Yes
Copy Jar files (third-party libraries, services, security, AdvWeb support	Yes	Yes	Yes	Yes
Copy EJB Jar file	Yes	Yes	Yes	
Copy Cluster EJB Jar			Yes	Yes
Add State Persistence			Yes	Yes
Copy Log Appender DLL (Steelhead)	Yes	Yes	Yes	Yes
Create a new logging directory	Yes	Yes	Yes	Yes
Setup Primary B2B Server Setup Secondary B2B Server			Yes Yes	Yes Yes
Create a report directory if nonexistent		Yes	Yes	Yes
Get revised Crystal Reports templates			Yes	Yes
Delete WebLogic Server Log	Yes	Yes	Yes	
Delete Logs		Yes	Yes	Yes
Delete Application Logs	Yes	Yes	Yes	
Expand Static content on Web Servers			Yes	Yes

1 – In the **Mydomain** developer environment, the AdvWeb War file is "unjared" and the files remain in an uncompressed format.

Deploying Builds to other Servers

After building, all the artifacts (WARs, JARs, class files, ears, etc) are placed in the directory D:\AdvWeb\MasterBuild\staging. The two exceptions are the **B2B server** and the **Crystal Reports**.

- The B2B server is built in the directory:
  D:\AdvWeb\dev\AdvWeb\build6.1\build\B2B and deployed to D:\B2B
- The Crystal Reports requires the directory D:\AdvWeb\Crystal\RPTFiles

File Directories used for Deployment

In the directories `MasterBuild\deployFalcon` and `MasterBuild\deployDeveloper` are several scripts to aid in deployment and running WebLogic.

- **`D:\MasterBuild\deployFalcon`** - contains all the scripts for Falcon staging machine.

- **`D:\MasterBuild\deployDeveloper\AdvanTg`** - contains all the scripts for the domain "`AdvanTG`" and server "`MyServer`." This mirrors the `Falcon` testing environment.

- **`D:\MasterBuild\deployDeveloper\mydomain`** - contains all the scripts the local machine for the domain "**`mydomain`**" and server "**`myserver`**". This is for the traditional development environment where AdvWeb is ***not deployed*** as WAR but in its normal format of unjared Web files, Wars, Jars, and EJBs.

Deployment Scripts

There are several deployment scripts written to aid in efficient deployment.

- **`D:\MasterBuild\deployFalcon\deployFalcon.cmd`** deploys all the artifacts in the staging directory to the WebLogic directory on `Falcon: d:\bea\wlserver6.1`.

  ⚠️ **Caution:** This script should be only executed on the Falcon server.

- **`D:\MasterBuild\deployPreProd\deployPreProd.cmd`** copies all the artifacts built on Falcon to the pre-production environment (Figure 1).

- **`D:\MasterBuild\deployPreProd\deployArchive.cmd`** copes all the artifacts of the specified labeled build to the pre-production environment (Figure 1).

- **`D:\MasterBuild\deployStaging\deployStaging.cmd`** copies all the artifacts built on Falcon to the Staging environment (Figure 1).

- **`D:\MasterBuild\deployStaging\deployArchive.cmd`** copies all the artifacts of the specified archive to the Staging environment.

- **`D:\MasterBuild\deployBeta\deployBeta.cmd`** copies all the artifacts built on Falcon to the Beta environment (Figure 1).

- **`D:\MasterBuild\deployBeta\deployArchive.cmd`** copies all the artifacts of the specified labeled build to the Beta environment.

- **`D:\MasterBuild\deployDeveloper\AdvanTG\deploy.cmd`** deploys all the artifacts in the `Falcon` "staging" directory to the `AdvanTG` domain directory on the local machine: `D:\WebGain\bea\wlserver6.1sp1\config\AdvanTG\applications`. This deploys AdvWeb as a War file. This assumes the domain **`AdvanTG`** and a server of **`MyServer`**.

- **`D:\MasterBuild\deployDeveloper\mydomain\deploy.cmd`** deploys all the artifacts in the `Falcon` "staging" directory to the `mydomain` directory on the local machine: `D:\WebGain\bea\wlserver6.1sp1\config\mydomain\applications`. This deploys AdvWeb exploded, not as a War. This assumes the domain **`mydomain`** and a server name of **myserver**.

- **`D:\MasterBuild\deployDeveloper\mydomain\debugweblogic.cmd`** runs WebLogic in debug mode. WebLogic 6.1 requires JDK 1.3.1 to execute. Debugging requires starting WebLogic with some debug flags and the port number.

The port specified in the **`debugweblogic.cmd`** script is number 4666. From Visual Café, choose the **File** menu and then the **Attach to Process** command. Type **`localhost:4666`** in the field named `Non-Registered VM`, to debug.

Building AdvanTG Web on Falcon

Key Files Used to Control the Build

For each environment, the names of the key directories used in the build and application properties are found in these files:

- `D:\AdvWeb\MasterBuild\Deploy<target>\`**`advweb.properties`**
- `D:\AdvWeb\MasterBuild\Deploy<target>\`**`deploy.properties`**
- `D:\AdvWeb\MasterBuild\`**`build.xml`**
- `D:\AdvWeb\MasterBuild\`**`buildinclude.properties`**
- `D:\AdvWeb\MasterBuild\`**`setantenv.cmd`**

Step-by-Step Procedure

These are the detailed steps that a Configuration/Release Manager would use to build an new version of AdvanTG Web on the development server called **`Falcon`**.

Procedure: Daily AdvanTG Web Build for Falcon

1) Send an email to these development/testing staff members:
   - Outlook group: **`SPEC: AdvanTG Web Dev`**
   - Eric Schneider
   - Penny Smith
   - Nazmul Khan

2) Announce that a new Falcon application build will begin in **15 minutes** so that all developers can check required source code into Visual SourceSafe. Remind developers to review their development checklist, prior to checking-in code. (See *Reference: Developer Pre-Build Checklist* on page 35.)

3) Use the Microsoft Terminal Services software to connect and login to the `Falcon` server. (See *Deploying a New Build to Staging* on page 20 for an example of this procedure.)

4) Using Visual SourceSafe, manually copy the latest versions of all AdvanTG Web build and deploy scripts with these instructions:
   — Double-click Visual SourceSafe
   — Highlight the top-level directory `$AdvWeb/MasterBuild`
   — Right-click to display the shortcut menu and choose the **Get Latest Version** command
   — Select the check boxes **Recursive**, **Make Writable**, and **Build Tree**
   — Click **OK**.

5) Using AdvanTG Web Visual SourceSafe repository, check to see if any files within these directories or their subdirectories were updated with any changes [Use **Show Differences** command] since the last build that might affect the application:

   - `D:\AdvWeb\Dev\`**`AdvWeb`**`\build6.1`
   - `D:\AdvWeb\Dev\`**`Services`**`\build6.1`
   - `D:\AdvWeb\`**`MasterBuild`**
   - `D:\`**`Security\Phase 2`**`\TGSLC\build6.1`

6) Open a command-line window and type these commands at the prompt:
   ```
   D:
   cd \Advweb\MasterBuild
   ```

7) Label specific source code in Visual SourceSafe at the MasterBuild root level by typing the command: **`label TEST_mmddyyvv`**

   where **mm** = month, **dd** = day, **yy** = year, and **vv** = desired build version (usually 01, 02, etc.)

   > **Note:** Visual SourceSafe uses version numbers to keep track of every change you make to your files and projects. This gives you the ability to retrieve any version of a file or project. The most useful method is with *user-defined labels*. You can associate a label with any version of any file or project. A user-defined label can be a string of up to 31 characters.
   >
   > Any of the following are valid labels: "`1.0`", "`2.01b`", "`Final Beta`", and "`Approved for QA`". After you apply these labels, you can retrieve files associated with a particular state of your project from the **History** dialog box, which is accessed from the **Show History** command on the **Tools** menu.
   >
   > You will probably find that at a project level, you refer to user-defined labels, and not internal version numbers. On the other hand, it is rare to label individual files. When you label a project with a descriptive text string, all the files in that project and subproject inherit the label.

8) Once the source code is labeled, extract the label source code and archive it with a similar command that uses the same label found in the previous step:
   ```
   extractLabelAndArchive TEST_mmddyyvv
   ```

   > **Caution:** This command is only run on Falcon.

9) Run the build using the command **`extractLabelAndArchive TEST_mmddyyvv`**:
   If the build is successful, send an email with this subject line to each of the parties listed in step 1 of this procedure:
   `B2B and myserver services on falcon will be rebooted in 5 minutes`
   (If the build fails, see *Troubleshooting Common Problems* on page 19.)

10) On Falcon, use the Control Panel to display the Services. Stop the services with these names in the following order: (See Table 2: Which Servers Run Specific AdvanTG Web Services page 13 for list of all services.)
    `RMI Service` (also stops B2B Service)
    `IIS Administration Service` (also stops WWW Publishing Service)
    `myserver` (WebLogic)

11) Using an editor (WebGain or UltraEdit), change the file (shown in Figure 3) named
    `D:\AdvWeb\dev\AdvWeb\Webroot\WEBINF\config\log4J-config.xml` and turn-on logging within the `AdvWeb.WAR` deployed.

Figure 3: log4j-config.xml

```
<appender name="WLSDebug" class="org.apache.log4j.FileAppender">
      <param name="File"   value="./logs/advweb-debug.log" />
      <param name="Append" value="true" />
      <layout class="org.apache.log4j.PatternLayout">
             <param name="ConversionPattern" value="%d %t %-5p %c{2} - %m\n"/>
      </layout>
     <param name="Threshold" value="DEBUG" />
</appender>
```

12) Type these commands: to change to the deployFalcon subdirectory and run the Falcon server deployment command.
    cd deployFalcon
    deployFalcon
    This command copies the application WARs, JARs, EJBs, B2b configuration files to appropriate WebLogic application and configuration directories. It also prepares logging directories and copies static content to the web server's root directories.

13) Using an editor (WebGain or UltaEdit), change the file named
    D:\B2B\config\logging.xml and add **D:** to the path for the B2B log file.

> **Note:** The previous step is necessary to prevent problems with machines that have disk partitioning into C: and D: logical devices.

14) Return to the Control Panel to display the **Services**. Start the services with the names:
    myserver
    B2B

15) Return to the Control Panel to display the **Services**. Start the services with these names in the following order: (See Table 4: AdvanTG Web Server Application Services on page 33 for list of all services.)
    B2B Service (alo starts **RMI Service**)
    World Wide Web Publishing Service (starts **IIS Administration Service)**
    myserver (WebLogic)

16) Test the AdvanTG Web application with an Internet Browser and the initial URL:
    https://adv2-dev.tgslc.org/TGSLC/portal.jsp

17) Login to the application and verify some default functionality
    (for example, **List Institutions**) is operational.

18) Send an email with this subject line to each of the parties listed in step 1 of this procedure:
    Falcon is back up with the new build labeled TEST_mmddyyvv

19) Log off Falcon and quit Terminal Services.

Troubleshooting Common Problems

Below are listed some of the common difficulties associated with building the AdvanTG Web application and the steps to correct the problem.

Incorrect Visual SourceSafe Permissions

At very beginning of build, it fails with message stating that the account doesn't have permissions to check out Visual SourceSafe files.

Procedure: Setting Visual SourceSafe Admin Account Permissions

1) Check the `/AdvWeb/MasterBuild/buildinclude.properties` file to ensure that the **vssafe.userid** and **vssafe.password** are set correctly. (See *Administrative Security* on page 10.)

2) Check with Mitchell Reid (extension 4657) to ensure that the VSS administrative account has the sufficient permissions.

Crystal Reports Files in Use

When the build tries to delete the Crystal Reports files, it fails, and returns a message saying the Crystal Reports files are in use.

Procedure: Dealing with Parallel Crystal Service

1) Shut down the `Parallel Crystal` service in the **Services** control panel.

2) Right-click the task bar to display the **Windows Task Manager** and shut down any processes which begin with the letters **pcre**.(For example, `pcreserver.exe`, `pcregateway`). There will usually be at least two processes and possibly more when Crystal Reports hangs.

3) Run the daily build process procedure: (See *Building AdvanTG Web on Falcon* on page 16.)

4) Restart the Parallel Crystal Services using the Control Panel, Services program.

Deploying a New Build to Staging
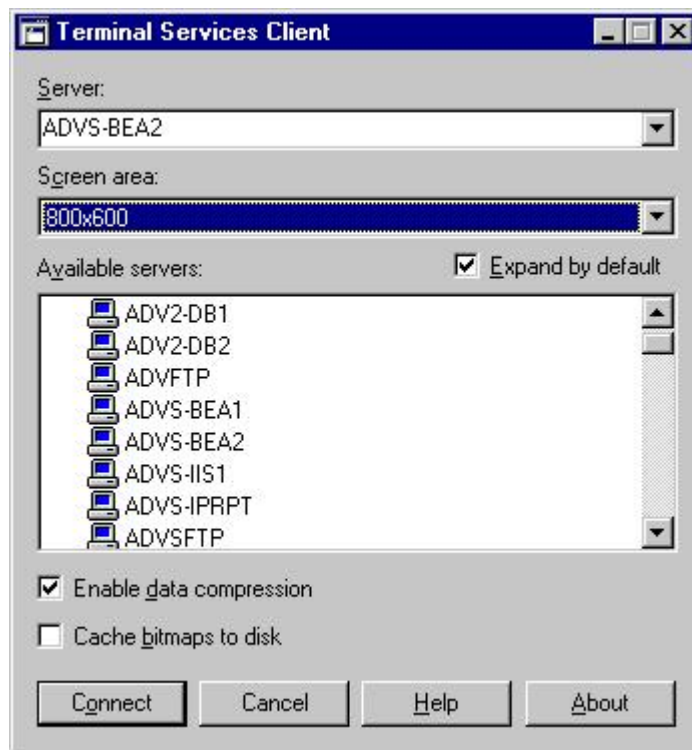
Deploy to Staging Procedure

Procedure: Deploying a Build to Staging

1) Double-click on the **Terminal Services** icon.

2) Select the `ADVS-BEA2` server from the **Terminal Services Client** window.

Figure 4: Terminal Services Client window

3) Login with a valid Administrator userID and password.
   (See *Security Required to Deploy AdvanTG Web* on page 8.)

4) Using this menu path, display the Services running on `ADVS-BEA2`:
   Start **»** Programs **»** Administrative Tools **»** Services**.**

5) Highlight each of these services on the appropriate Staging machine:
   RMI Service (also stops B2B Service)
   IIS Administration Service (also stops WWW Publishing Service)
   BEA WebLogic **AdvStage** (WebLogic)
   adv2stagewladmin   *(Only on the ADVS-BEA1 server!)*
    (You will be warned that other services will be stopped. Press the **Yes** button to proceed.)

6) Right-click the mouse and select the **Stop** command for each of the four services.

Figure 5: ADVS-BEA2 Services window
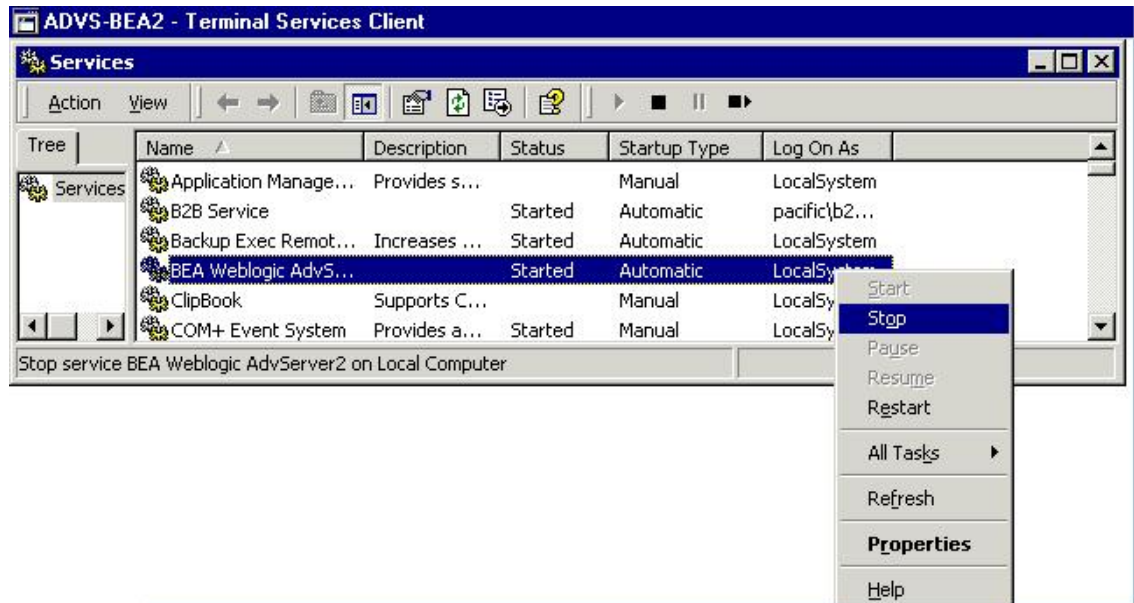
7) Minimize the instance of the `ADVS-BEA2` server Terminal Services Client.

8) Double-click once more the **Terminal Services** icon.

9) Select the server `ADVS-BEA1` from the display list in **Terminal Services Client** window.

10) Login with a valid Administrator userID and password.

11) Using this menu path, display the Services running on `ADVS-BEA1`:
    Start **»** Programs **»** Administrative Tools **»** Services**.**

12) Highlight each of these services:
    ```
    adv2stagewlserver1
    RMI Service
    adv2stagewladmin
    ```
    (Only on the `ADVS-BEA1` server!)
    (You will be warned that other services will be stopped. Press the **Yes** button to proceed.)

13) Right-click the mouse and select the **Stop** command. (See Figure 5.)

14) Double-click once more the **Terminal Services** icon.

15) Select the server `ADVS-IIS1` from the display list in **Terminal Services Client** window.

16) Login with a valid Administrator userID and password.

17) Using this menu path, display the Services running on `ADVS-BEA1`:
    Start **»** Programs **»** Administrative Tools **»** Services**.**

18) Highlight each of these services:
    ```
    IIS Admin Service
    ```
    (You will be warned that other services will be stopped. Press the **Yes** button to proceed.)

19) Right-click the mouse and select the **Stop** command. (See Figure 5.)

Procedure: Preserving the B2B Folder Contents before a New Build

When the current `DeployStaging` script runs, it destroys all the contents within the B2B
folders on the AdvanTG WebLogic servers. To prevent this destruction of B2B files, a script
needs to run, prior to running ther `DeployStaging` script.

20) Run the `B2BCopy.cmd` script to archive the B2B folder contents (including subfolders) to a directory on the `advsftp` server. The folder name is automatically determined by the system date and the number of times the script was previously run that day. See

Figure 6: B2BCopy Script

```
@echo off

setlocal

SET B2B_1=d:\b2b\test
SET B2B_2=\\advs-bea2\b2b\test
SET B2B_TARGET=\\advsftp\B2B_backup


rem Get the system date and check to see how many backups have been done
for /f "tokens=1-4 delims=/ " %%a in ('date /t') do (set weekday=%%a& set date=%%b&
set month=%%c& set year=%%d)

set DATECOUNTER=0
SET DATEPATH=%date%%month%%year%
rem echo %DATEPATH%
:LOOP
if exist %B2B_TARGET%\bea1\%DATEPATH%%DATECOUNTER%\ goto ADD
goto COPYFILES
:ADD
set /a DATECOUNTER+=1
rem echo LOOPING
rem echo %DATECOUNTER%
goto LOOP

:COPYFILES
Echo Copying B2B content

SET DATEPATH=%date%%month%%year%%DATECOUNTER%

xcopy /R /Y /E %B2B_1% %B2B_TARGET%\bea1\%DATEPATH%\
xcopy /R /Y /E %B2B_2% %B2B_TARGET%\bea2\%DATEPATH%\

echo %DATEPATH%
pause
```

⚠️ **Caution:** Only skip step 20 if you are sure the testing team has no need for the data within the B2B folders on the two WebLogic Staging servers.

21) Make sure you have extracted the latest `deployStaging.cmd` command file from Visual SourceSafe.

22) Run the deploy script -> **D:\ADVWEB\deployStaging.cmd**.
It takes between **five** to **ten** minutes to copy all files from `Falcon`.

23) Use the Terminal Services Client to restart these services on `ADVS-BEA1` server:
B2B Service
`World Wide Web Publishing Service` (on `ADVS-IIS` server)
`adv2stagewladmin`
`adv2stagewlserver1`

24) Highlight each service in the **Services** window, right-click the mouse, and select the **Start** command from the shortcut menu.

25) Use the Terminal Services Client to restart these services on `ADVS-BEA2` server:
`B2B Service`

```
adv2stagewlserver2
```

26) Highlight each service in the **Services** window, right-click the mouse, and select the **Start** command from the shortcut menu.

> **Note:** It will take a few minutes for all the services to resume operation

27) Test the staging IIS5 server by accessing the initial Java Server Page using the URL: **https://adv2-stage.tgslc.org/TGSLC/portal.jsp**

28) Test the staging WebLogic application server by logging-in with a test userID and performing an AdvanTG Web operation, such as List Institutions. Page through one or more screens.

29) Send out an email to these mailing groups indicating the new build version label:
   - Outlook group: **SPEC: AdvanTG Web UAT Team**
   - Outlook group: **SPEC: AdvanTG Web Test Team**

The email might contain the text: " Staging back up with build labeled TEST_07120201"

30) Tell Anil Gari/Wasif Awan (extension 4416) to add the build version label to the TestDirector defect tracking application.

General Staging Troubleshooting

Most difficulties after a staging deployment involve one or more services that have not started on the staging servers. Using the Terminal Services client, you can login to each machine, examine each service, and manually start the service if necessary.

Examine the services on these servers to ensure they are correctly running:
ADVS-IPRPT – LDAP/Crystal Reports Server
ADVS-IIS1 – Microsoft Internet Information Server

Services on Other Staging Servers

Procedure: Checking and Restarting other Services

1) Login to ADVS-IPRPT with Terminal Services.

2) Using the Start Menu **»** Programs **»** Administrative Tools **»** Services command, display the status of these services:
   Parallel Crystal
   iplanet Admin
   iplanet Server

3) Start any service by highlighting the service, right-clicking, and using the **Start** command..

4) Logoff the ADVS-IPRPT server and login to the IIS Web Server named ADVS-IIS1.

5) Using the Start Menu **»** Programs **»** Administrative Tools **»** Services command, display the status of these services:
   IIS Admin Service
   World Wide Web Publishing

6) Start any service by highlighting the service, right-clicking, and using the **Start** command.

7) Logoff the `ADVS-IIS1` Staging Web Server.

Changes to `AdvWeb.properties` file

The `AdvWeb.properties` file is unique to each server within the AdvanTG configuration. If the logs show a message similar to "`variable not defined…`", check with Travis Bowen (ext. 4331) or Sushil Sureka (ext. 4479) about this file.

Other Problems

For other questions or problems, please call Bob Blackard at extension 4823

Deploying Builds to Pre-Production

Deploying builds to Pre-Production is similar to deploying a build the Staging environment. In this procedure we need to work with four servers simultaneously, instead of the three servers in Staging.

Since `Libra1` and `Libra2` replicate each other's changes, the implications for deployment include:

- Stop the `BEA Admin Server` service on Libra1 as your final action before deployment.
- Deploy the AdvanTG Web application only to `Libra1`. It will be automatically replicated on `Libra2`. (See *Deploying the Application* on page 25.)
- Start the services using the order specified in *Starting Pre-Production Services* on page 26.

Stopping Pre-Production Services

Procedure: Stopping Services

1) Open Terminal Services clients for the servers: `Libra1`, `Libra2`, `Hydra1` and `Hydra2`. (See *Deploy to Staging Procedure* on page 20.)

2) Stop the services on each machine by:
   —Right clicking the **My Computer** icon and select the **Manage** command.
   —Expanding the **Services** tree node, followed by the **Applications** node.
   —Selecting the **Services** node. (This is the Services manager.)

3) For `Hydra1` and `Hydra2`, stop:
   `IIS Admin Service`, which stops the `World Wide Web Publishing Service` also.

4) For `Libra1` and `Libra2`, stop:
   `RMI Service`, which stops the `B2B Service` also.

5) For `Libra2`, stop the `BEA AdvServer2` service.

6) For `Libra1`, stop the `BEA AdvServer1` service.

7) For `Libra1`, stop the `BEA AdminServer` service.

Deploying the Application

> **Caution:** Prior to running the script, verify that you have the latest file version from the Visual SourceSafe folder: `$AdvWeb/MasterBuild/deployPreProd.cmd`

Procedure: Deploying AdvanTG Web to Pre-Production Environment

8) Get the latest version of the `deployPreProd.cmd` command script and place it on Libra1 in the folder **D:\ADVWEB\deployPreProd.cmd**.

9) On `Libra1`, right-click **My Computer** and select **Explore** (this is the Explorer).

10) Expand the `D:` logical node.

11) Select the D:\Deploy Script folder.

12) Double-click to run the `deployPreProd.cmd` command script

Starting Pre-Production Services

Pre-production services are generally restarted in the ***exact opposite order*** in which they were stopped (with the exception of the IIS5 Web Servers).

Procedure: Restarting Pre-production Services

1) For `Hydra1` and `Hydra2`, start:
   `World Wide Web Publishing Service` which also starts the `IIS Admin Service`.

2) For `Libra1` and `Libra2`, start:
   `B2B Service`, which also starts the `RMI Service`.

3) For `Libra1`, start the `BEA AdminServer` service.

4) On `Libra1`, select the Explorer window and perform these actions:
   — select the `D:` logical node.
   — expand the tree selecting the folders `bea`, `wlserver6.1`, `config`, and `AdvanTG` as illustrated in Figure 7 Folder Path to BEA WebLogic Logs folder.
   — select the `logs` folder.

Figure 7: Folder Path to BEA WebLogic Logs folder
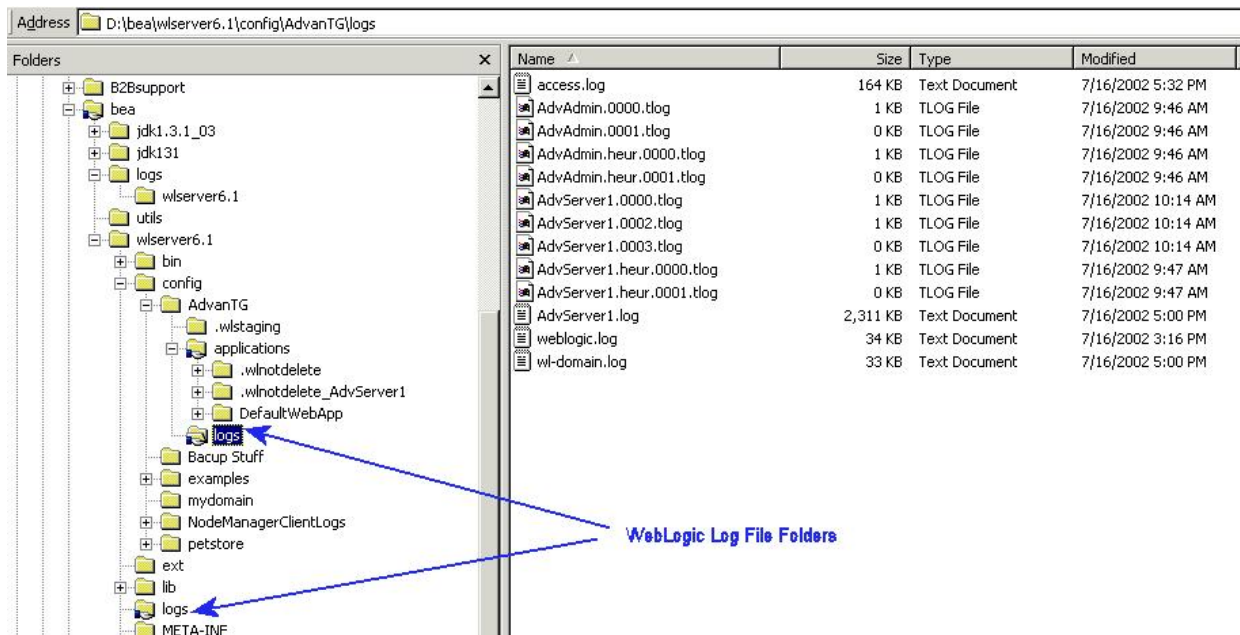
5) Monitor the `wl-domain.log` file until it's size reaches 1K bytes.

6) Double-click `wl-domain.log` and check for content similar to Figure 8.
   Once this content displays, the AdminServer service has successfully started.

Figure 8: wl-domain.log – Admin Server Started Successfully

```
####<Jun 21, 2002 2:46:05 PM CDT> <Notice> <WebLogicServer> <libra1> <AdvAdmin>
<main> <system> <> <000327> <Starting WebLogic Admin Server "AdvAdmin" for domain
"AdvanTG">
####<Jun 21, 2002 2:46:10 PM CDT> <Notice> <Management> <libra1> <AdvAdmin> <main>
<system> <> <141053> <Application Poller not started for production server.>
####<Jun 21, 2002 2:46:10 PM CDT> <Notice> <WebLogicServer> <libra1> <AdvAdmin>
<ListenThread> <system> <> <000201> <ListenThread listening on port 7003, ip address
172.16.10.108>
####<Jun 21, 2002 2:46:11 PM CDT> <Notice> <Management> <libra1> <AdvAdmin> <main>
<system> <> <141030> <Starting discovery of Managed Server... This feature is on by
default, you may turn this off by passing -Dweblogic.management.discover=false>
```

7) For `Libra1`, start the `BEA AdvServer1` service.

8) For `Libra2`, start the `BEA AdvServer2` service.

9) On `Libra1`, select the Explorer window and monitor the `wl-domain.log` file until it's size increases to approximately 3K bytes.

10) Double-click `wl-domain.log` and check for content similar to Figure 9. Once this content displays, the BEA WebLogic Cluster has successfully started.

Figure 9: wl-domain.log – Cluster Completely Started

```
####<Jun 21, 2002 2:50:34 PM CDT> <Notice> <WebLogicServer> <libra1> <AdvServer1>
<ExecuteThread: '8' for queue: '__weblogic_admin_rmi_queue'> <system> <> <000328>
<Starting WebLogic Managed Server "AdvServer1" for domain "AdvanTG">
####<Jun 21, 2002 2:50:38 PM CDT> <Notice> <WebLogicServer> <libra2> <AdvServer2>
<ExecuteThread: '8' for queue: '__weblogic_admin_rmi_queue'> <system> <> <000328>
<Starting WebLogic Managed Server "AdvServer2" for domain "AdvanTG">
####<Jun 21, 2002 2:51:41 PM CDT> <Notice> <WebLogicServer> <libra1> <AdvServer1>
<ExecuteThread: '8' for queue: '__weblogic_admin_rmi_queue'> <system> <> <000201>
<ListenThread listening on port 7001, ip address 172.16.10.7>
####<Jun 21, 2002 2:51:43 PM CDT> <Notice> <Cluster> <libra1> <AdvServer1>
<ExecuteThread: '8' for queue: '__weblogic_admin_rmi_queue'> <system> <> <000102>
<Listening for multicast messages (cluster AdvCluster1) on port 7001 at address
237.20.21.11>
####<Jun 21, 2002 2:51:43 PM CDT> <Notice> <WebLogicServer> <libra1> <AdvServer1>
<ExecuteThread: '8' for queue: '__weblogic_admin_rmi_queue'> <system> <> <000330>
<Started WebLogic Managed Server "AdvServer1" for domain "AdvanTG" running in
Production Mode>
####<Jun 21, 2002 2:51:45 PM CDT> <Notice> <WebLogicServer> <libra2> <AdvServer2>
<ExecuteThread: '8' for queue: '__weblogic_admin_rmi_queue'> <system> <> <000201>
<ListenThread listening on port 7001, ip address 172.16.10.61>
####<Jun 21, 2002 2:51:46 PM CDT> <Notice> <Cluster> <libra2> <AdvServer2>
<ExecuteThread: '8' for queue: '__weblogic_admin_rmi_queue'> <system> <> <000102>
<Listening for multicast messages (cluster AdvCluster1) on port 7001 at address
237.20.21.11>
####<Jun 21, 2002 2:51:46 PM CDT> <Notice> <WebLogicServer> <libra2> <AdvServer2>
<ExecuteThread: '8' for queue: '__weblogic_admin_rmi_queue'> <system> <> <000330>
<Started WebLogic Managed Server "AdvServer2" for domain "AdvanTG" running in
Production Mode>
```

11) Test the Pre-prod IIS5 web server by accessing the initial Java Server Page using the URL: **https://adv2-prod.tgslc.org/TGSLC/portal.jsp**

12) Test the Pre-Prod WebLogic application server by logging-in with a test userID and performing an AdvanTG Web operation, such as List Institutions. Page through one or more screens.

Preserving WebLogic Logs for Troubleshooting

During Performance Testing or difficulty with the WebLogic servers, it is useful to save the logs generated by WebLogic. **Robert Blackard** has developed an easy-to-use, standard procedure for saving the logs.

Procedure: Saving WebLogic Logs for Evaluation

1) **Stop all services on all the servers**, except the `RMI Service` on `Libra1` and `Libra2`. Refer to the procedure in the section *Stopping Pre-Production Services* on page 25.

2) Determine a time stamp for saving these logs in the format - **`MMddyyyyhhmm`**, where:
   `MM` — numeric month of the year with a leading zero (01-12)
   `dd` — numeric day of the month (01-31)
   `yyyy` — four digit year (2000-2099)
   `hh` — the hour of the day in military time (0-23)
   `mm` — the minute of the hour -(0-59)

3) On `Libra1` and `Libra2`:
   — right-click the **My Computer** icon and select the **Explore** (Explorer application).
   — select the `D:` logical node.
   — expand the tree selecting the folders `bea`, `wlserver6.1`, `config`, and `AdvanTG` as illustrated in Figure 7: Folder Path to BEA WebLogic Logs folder.
   — select the `logs` folder.

4) On `Libra1` and `Libra2`, continue with these steps:
   — right-click in an open space on the right panel of the Explorer.
   — select the **New** command from the shortcut menu and **Folder** from the cascading menu.
   — label the folder using the timestamp created in **step two**.)

5) On `Libra1` and `Libra2`, continue with these steps:
   — move all files (but not folders) from `logs` into the newly created folder.
   — select the `logs` folder of `D:\bea\wlserver6.1` (up two levels in the folder tree)
   — move the files in the `D:\bea\wlserver6.1\logs` folder into the created folder.

6) On `Hydra1` and `Hydra2`:
   — right-click the My Computer icon and select the Explore (Explorer application).
   — expand the `D:` logical node.
   — expand the tree selecting the folders `Web Logs`, `AdvanTG Production Web Site`, and `W3SVC3` as illustrated in *Figure 10: Folder Path to IIS5 Logs folder*.
   — select the `W3SVCS` folder.

Figure 10: Folder Path to IIS5 Logs folder
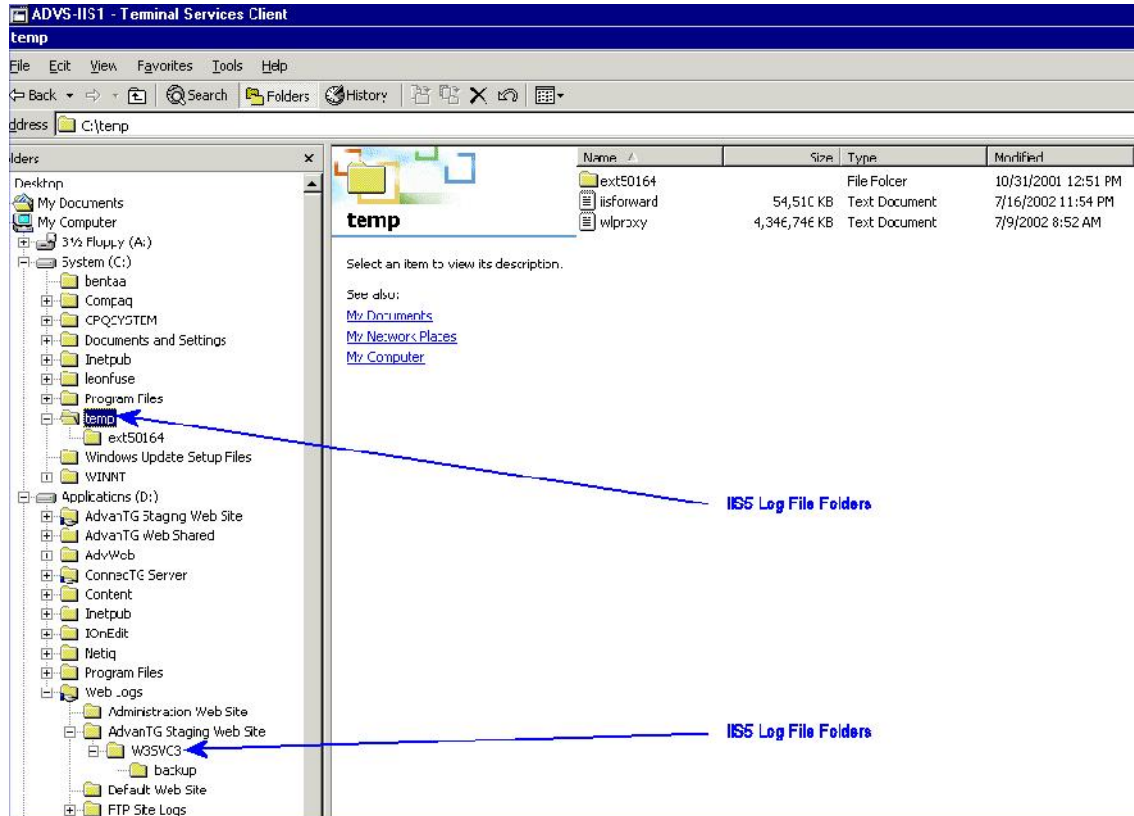
7) On Hydra1 and Hydra2, continue with these steps:
   — right-click in an open space on the right panel of the Explorer.
   — select the **New** command from the shortcut menu and **Folder** from the cascading menu.
   — label the folder using the timestamp created in **Step two**.

8) On Hydra1 and Hydra2, continue with these steps:
   — move all files (but not folders) from W3SVCS into the newly created folder.
   — expand the C: logical node.
   — select the TEMP folder.

9) On Hydra1 and Hydra2, pressing the [**Shift**] key down, move these files into the created folder on the D: logical unit.
   iisforward.log
   wlproxy.log

⚠ **Caution:** The [**Shift**] key moves the files instead of creating copies of the files.

10) Start all services on the servers except the B2B Service on Libra1 and Libra2.
    Refer to the procedure in section *Starting Pre-Production Services* on page 26.

Deploying AdvanTG Web to Development Servers

AdvanTG and mydomain domains

- **D:\MasterBuild\deployDeveloper\AdvanTG\deploy.cmd** deploys all the artifacts in the `Falcon` "staging" directory to the `AdvanTG` domain directory on the local machine: `D:\WebGain\bea\wlserver6.1sp1\config\AdvanTG\applications`. This deploys AdvWeb as a War file. This assumes the domain **AdvanTG** and a server of **MyServer**.

- **D:\MasterBuild\deployDeveloper\mydomain\deploy.cmd** deploys all the artifacts in the `Falcon` "staging" directory to the `mydomain` directory on the local machine: `D:\WebGain\bea\wlserver6.1sp1\config\mydomain\applications`. This deploys AdvWeb exploded, not as a War. This assumes the domain **mydomain** and a server name of **myserver**.

GUI Server

Procedure: Deploy build to the Developer's GUI Server

1) Stop the `GUI server` using the Services management center:
   — Right-click the **My Computer** icon and select the **Manage** command.
   — Expand the **Services and Management** node to select **Services**.
   — Scroll to the **guiserver** service and click **Stop** button.

2) Copy the WebLogic AdvWeb Entity Bean Jar file to the GUI development directory:
   — Right-click the **My Computer** icon and select the **Explore** command.
   — Expand the D:\ directory to display the directories under `D:\bea`
   — Copy the Jar file `bea/wlserver6.1/config/AdvanTG/Applications/`**wls_ejb_AdvwebEntityBean** to the directory `bea/wlserver6.1/config/GuiDev/Applications`

3) Extract the contents of the `AdvanTG/AdvWeb.war` to the folder `GuiDev/Applications/AdvWebGuiDev`. Right-click and use the **WinZip** command.

4) Extract the contents of the `AdvanTG/TGSLC.war` to the folder `GuiDev/Applications/TGSLCGuiDev`. Right-click and use the **WinZip** command.

5) Delete all logs in the `GuiDev/Logs` folder.

6) Add `.\config\GuiDev` to the logging configuration file in two places: `GuiDev\Applications\AdvWebGuiDev\WEB-INF\config\log4j-config.xml`

7) Add `.\config\GuiDev` to the logging configuration file in three places: `GuiDev\Applications\TGSLCGuiDev\WEB-INF\config\log4j-config.xml`

8) Add "**GuiDev**" to "**AdvWeb**" to the `javaclass` section at end of the file: `GuiDev\Applications\TGSLCGuiDev\WEB-INF\SSOWPConfig.xml`

9) Crosscheck AdvWeb.properties in wlserver6.1/config/GuiDev/advweb.properties and wlserver6.1/config/AdvanTG/advweb.properties. Copy missing lines from AdvanTG over to GuiDev.

10) Restart the `GUI server` using the Services management center:
   — Right-click the **My Computer** icon and select the **Manage** command.
   — Expand the **Services and Management** node to select **Services**.
   — Scroll to the **guiserver** service and click **Start** button.

Deploying an Archived Build to Staging

Each AdvanTG Web build is archived to the `ADVSTFP` server and stored under a file folder in the directory – `C:\Adv-Builds\TEST_LABEL`.

Figure 11: Archived AdvanTG Web Builds
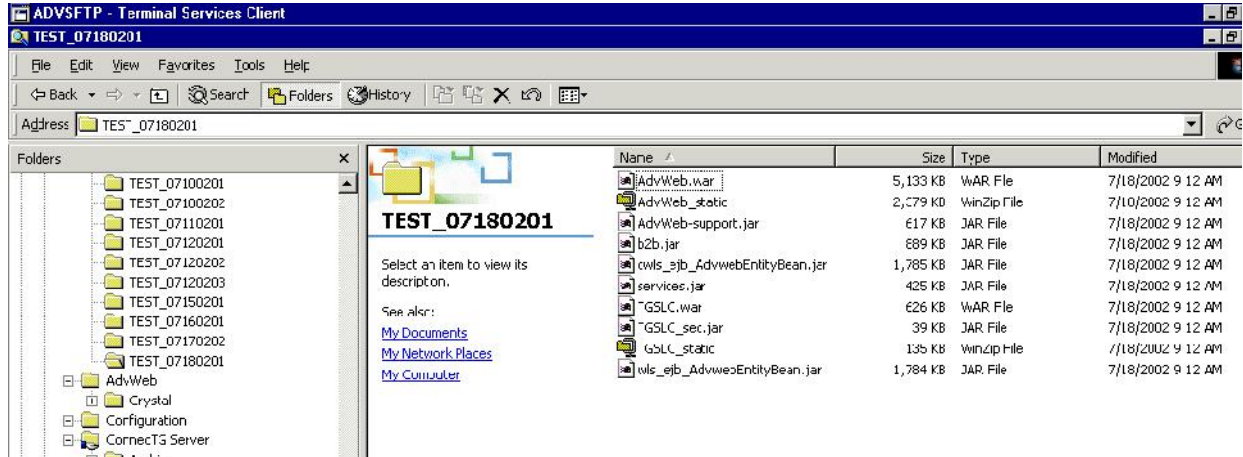
Procedure: Retrieve and Deploy an Archived Build to Staging

1) Stop the staging services as shown in the section *Deploy to Staging Procedure* on page 20.

2) Get the latest version of all deployment scripts from this folder in Visual SourceSafe:
   `$AdvWeb/MasterBuild/deployStaging/`

3) Run the script named deployArchive and provide the version label as the parameter.
   For example, you would type the commands:
   `cd D:\AdvWeb\MasterBuild\deployStaging`
   `deployArchive TEST_07180201`

4) Restart the staging services as shown in the section *Deploy to Staging Procedure* on page 20.

Reference: Services on AdvanTG Web Servers

This table is provided for experienced system administrators who often need to stop and start application services as part of the AdvanTG Web build and deployment.

Table 4: AdvanTG Web Server Application Services

Machine Name	AdvanTG Machine Function	Services on this Machine
Falcon	**Build Machine**	B2B Service Parallel Crystal myserver - Weblogic[1] IIS Admin Service World Wide Web Publishing Service
ADVS-BEA1	**Staging** – WebLogic Master **Beta** – WebLogic lave	adv2stagewladmin B2B Service RMI Service Steelhead adv2stagewlserver1 adv2betawlserver1
ADVS-BEA2	**Staging** – WebLogic Slave **Beta** – WebLogic Master	B2B Service RMI Service adv2stagewlserver2 Steelhead adv2betawladmin adv2betawlserver2

ADVS-IPRPT	**Staging** – LDAP/Crystal Report
	
	Beta – LDAP/Crystal Report
		iPlanet Administrator Server 5
		iPlanet Directory Server 5
		    (advs-iprpt.tgslc.org)
		iPlanet Directory Server 5
		    (advs-iprpt2)
		Parallel Crystal

ADVS-IIS1	**Staging** – IIS5 Web Server	IIS Admin Service World Wide Web Publishing Service
ADVSFTP	**Staging** – MORAY Simulation	FTP publishing Service IIS Admin Service Sockeye Steelhead
Virgo1	**Pre-Prod** – LDAP iPlanet	iPlanet Administrator Server 5 iPlanet Directory Server 5 (pre-prod.tgslc.org)
Virgo2	**Pre-Prod** – LDAP iPlanet	iPlanet Directory Server 5 (pre-prod.tgslc.org)

Libra1	**Pre-Prod** – WebLogic TopLink Master	AdvAdmin B2B Service RMI Service BEA Weblogic AdvServer1
Libra2	**Pre-Prod** – WebLogic TopLink Slave	B2B Service RMI Service BEA Weblogic AdvServer2 Steelhead
Hydra1	**Pre-Prod** – IIS5 Web Server	IIS Admin Service World Wide Web Publishing Service
Hydra2	**Pre-Prod** – IIS5 Web Server	IIS Admin Service World Wide Web Publishing Service

1 World Wide Web Publishing Service depends upon the IIS Admin Service.

2 The B2B Service depends upon the RMI Service.

Reference: Developer Pre-Build Checklist

Prior to checking source code into Visual SourceSafe to be used in a new build, developers should review the following checklist:

Completed	Pre-Build Developer Task
☐	Complete edits on required source code.
☐	Perform a **Get Latest Version** from the `$AdvWeb/development/` source trees.
☐	Get the latest version of the `$AdvWeb/MasterBuild` folder files
☐	Run the `Build` command
☐	Deploy the build to `mydomain`
☐	Start the Weblogic server within your development environment
☐	Login to the AdvanTG Web application
☐	Perform a simple operation, such as List Institutions. Page through several screens with the Next and Previous buttons.
☐	Check-in all necessary source code to Visual SourceSafe
☐	Notify the Build Manager of any changes made to: `$AdvWeb/MasterBuild` folder files or `AdvWeb.properties` files
☐	_____
☐	_____