

ECaccess User Guide

User Support

Operations Department

Version 4.0.1

March 2013



©Copyright 2013

European Centre for Medium-Range Weather Forecasts
Shinfield Park, Reading, RG2 9AX, United Kingdom

Literary and scientific copyrights belong to ECMWF and are reserved in all countries.

The information within this publication is given in good faith and considered to be true, but ECMWF accepts no liability for error, omission and for loss or damage arising from its use.

Contents

1	Introduction	2
2	Ecaccess concepts	3
2.1	ECaccess gateway	4
2.2	Using an ECaccess gateway	4
2.3	Plugins	4
3	Security authentication	6
4	Unattended file transfer - ectrans	7
4.1	Target location	7
4.2	ectrans command	7
4.2.1	Transfer to a Member State host via gateway	9
4.2.2	Transfer from a Member State host via gateway	9
5	Web Toolkit - The full featured client	10
5.1	Certificate management	10
5.2	General information	11
5.3	Gateway information	12
5.4	File management	12
5.5	Batch job management	13
5.6	Management of events	14
5.7	Management of ECtrans transfers	14
5.8	Management of ECtrans associations	14
5.9	Execution return codes	15
6	Web Services - Standard thin clients	15
6.1	Obtaining CURL	15
6.2	Environment	16
6.3	Input/output	16
6.4	Authentication	16
6.4.1	Certificates	16
6.4.2	Tokens	17
6.5	Cosinfo	17
6.6	Gateways	18
6.7	Operations	19
6.8	Associations	20
6.9	Protocols	22
6.10	Files	23
6.11	Jobs	26
6.12	Events	29
6.13	Queues	31
6.14	Transfers	32
7	The Web server	35
7.1	Authentication	35
7.2	Features	35
7.3	Users views	37

7.4	Ectrans setup	42
7.5	NX service	46
7.5.1	How to connect using a web browser	46
7.5.2	Example of session starting a standalone xterm on the supercomputer	47
7.5.3	Example of session starting a virtual desktop on ecgate	48
7.5.4	How to connect using a standalone NX client	49
7.5.5	WindowMaker overview	52
8	Monitoring tools	54
8.1	Monitoring batch job submissions	54
8.2	Monitoring ectrans file transfers	56
9	The Telnet server	57
10	The SSH server	58
11	X11 connections	59
11.1	Starting xterm within a SSH session	59

The ECaccess change history can be found at www.ecmwf.int/services/ecaccess/download/changelog.html

1 Introduction

The ECaccess software gives Member States ¹ and other ECMWF users batch and interactive access to the ECMWF computing and archiving facilities. Access is available via the Internet as well as via RMDCN.

This user guide, which is intended for all users of the ECaccess software, describes the concepts and procedures for accessing data and running jobs at ECMWF. If you are to perform the administrative task of installing and/or maintaining the ECaccess software, you should study the *ECaccess Administrator's Guide* (see <http://www.ecmwf.int/services/ecaccess/download/>). For the gateway concepts and procedures see section 2.

This guide is structured as follows:

Getting started

- Section 2 describes the ECaccess global architecture, focusing on the FTP, Web, Telnet and X11 components.
- Section 3 gives an overview on interactive and shell script user authentication.

Running batch work at ECMWF

For automating data transfers and submitting batch jobs, refer to:

- Section 4 describes how to initiate unattended transfers from ECMWF.
- Section 5 describes the ECaccess Web Toolkit, the full-featured client
- Section 6 describes the ECaccess Web Services, in particular how to CURL to interact with the ECaccess Web Services API

ECaccess on-line

For access to on-line ECMWF computing facilities, refer to:

- Section 7 describes web-based management of jobs and file transfers to and from the ECHOME (for "ecgate" home directory), ECSCRATCH (for "ecgate" scratch directory) and ECFS directories.
- Section 8 describes web-based monitoring and trouble- shooting of batch jobs and file transfers.
- Section 9 describes logging in at ECMWF via the gateway's single-sign-on Telnet server component.
- Section 10 describes logging in at ECMWF via the gateway's single-sign-on SSH server component.
- Section 11 describes starting X11 applications on ECMWF servers using the single-sign-on X11 access component.

¹In the following "Member States" (MS) includes "Co-operating States".

2 Ecaccess concepts

ECaccess is a framework for batch and interactive access to ECMWF services for Member States and other ECMWF users.

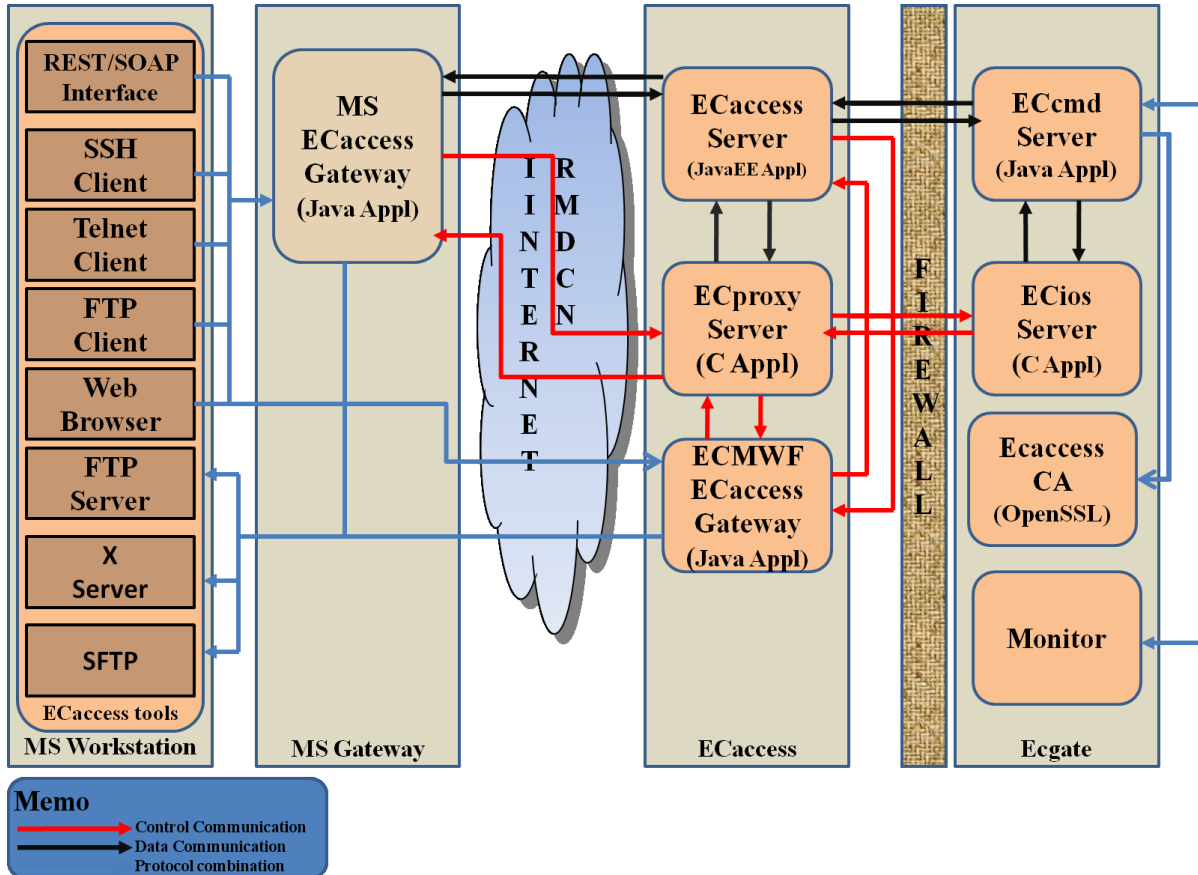


Figure 1: ECaccess design layout.

The components of ECaccess are:

- The ECaccess gateways: all Member State users can access the ECMWF computing and archiving facilities through a gateway. Full ECaccess functionality requires an ECaccess gateway to be installed at the Member State. Alternatively, reduced ECaccess functionality is available on the ECMWF ECaccess gateway.
- The ECaccess Server: all gateways are connected to this server. It provides technical and high level services to the gateway, allowing generic access to computing and archiving facilities at ECMWF (through "ecgate").
- The "ecgate" server: includes services such as the local LoadLeveler batch system, the (LoadLeveler) batch system on c2a (the High Performance Computing Facility) and access to the ECFS, HOME and SCRATCH storage areas.

To allow authentication and improve security, an ECaccess Certification Authority (ECCA) certifies all ECaccess components.

2.1 ECaccess gateway

The gateway software is provided for Member States' remote access to ECMWF computing and archiving facilities. Throughout the guide, the terms "gateway" and "ECaccess gateway" are used interchangeably. Gateways include a model for the management of "plugin" services. A plugin is a piece of code that handles requests/responses flowing through the gateway. Currently, there are plugins for incoming FTP, HTTP/S, X11, SSH and Telnet (MSgateways only) requests to ECMWF. Additional plugins are planned. On top of the SSH plugin the NX application can be used for interactive access to ECMWF. The ECMWF ECaccess gateway (hereafter referred to as "ECgateway") can be used on its own. Nevertheless, using a Member State ECaccess gateway (hereafter referred to as "MSgateway") instead offers the following features and advantages over using the ECgateway on its own:

- Secure tunnel between ECMWF and MSgateway: all services are channelled through SSL (Secure Socket Layer) secure connections to ensure data integrity. For confidentiality, administrators can set up encryption.
- Security authentication: protocols such as FTP or Telnet use only basic security mechanisms during their login process. The MSgateway plugins invoke an SSL protocol component for user authentication.
- Low resource usage / fast response: opening and closing SSL connections takes a significant amount of CPU time, bandwidth and memory. MSgateways maintain a set of permanent SSL connections (to the ECaccess server) for their plugins.
- Web memory cache: pages collected by the MSgateway from ECMWF and passed to Member State browsers can be stored in a memory cache. If the same page is required again, it is retrieved from this cache. Since this cache is located on the MSgateway, this is quicker than access through the Internet.

2.2 Using an ECaccess gateway

If the basic features, available via the ECMWF ECaccess gateway interfaces, are sufficient, you can use "ecaccess.ecmwf.int" for the web and the FTP interface. The Web Toolkit (section 5) use "ecaccess.ecmwf.int" as the default gateway name. If you have access to RMDCN and want to use it for accessing ECMWF, you can use "msaccess.ecmwf.int" instead.

If you wish to use the advanced features, only available via a Member State ECaccess gateway, you will need to find out, on which host this gateway has been installed at your local site and which FTP and HTTPS ports are being used by that gateway. You may be able to obtain this information by running the `ecaccess-gateway-name Web Toolkit` command. If `ecaccess-gateway-name` is in your command path, it will provide information about the ECaccess gateway you are using.

If the command is not available, you will need to contact your local ECaccess administrator or Computing Representative. You can also email advisory@ecmwf.int.

2.3 Plugins

By default, the following plugins are automatically started on all the gateways:

- The FTP plugin: allows Member State users to submit jobs and to transfer files (between their own computer on one side and ECMWF file systems and ECFS on the other side). This extended FTP server can also be used for access to ECMWF computing and archiving facilities from within shell scripts.
- The HTTP/S plugin: for job and file transfer management/monitoring from a browser.
- The Telnet and the X11 plugins (available on MSgateways only): provide access to ECMWF servers with a single-sign-on login process. Communication and authentication are established through the gateway.
- SSH is increasingly used for external connections. ECaccess includes an SSH plugin which will allow you to access ECMWF and run X11. Note that only SSH protocol version 2 is supported.

3 Security authentication

The gateways uses two built-in security mechanisms to control access to ECMWF:

- Interactive authentication: users will be prompted for their ECMWF user identifier and the PASSCODE (obtained by entering their PIN number into the security token).
- Batch authentication: users need to create an EAccess certificate before they access ECMWF facilities. This method allows Member State users to automate authentication within scripts. The HTTP/S, Telnet, X11 and SSH plugins support only the first method. The FTP plugin supports both.

The EAccess certificate is a standard X509 digital certificate saved on the user's computer as a file. It identifies a user to the gateway. The EAccess Certification Authority (ECCA) signs each certificate. Therefore, when a user provides his certificate to the gateway, its signature is checked using the ECCA public key for verification. A certificate can be created:

- Using the “ecaccess-certificate-create” command: this is described in section [5.1](#).
- Using the Web interface: login to the Web server (providing an ECMWF user identifier and token PASSCODE) and in the menu click the “Get Certificate” option to download the new Certificate, see section [7](#).

The EAccess certificate is valid for 7 days for all services.

4 Unattended file transfer - ectrans

The `ectrans` command allows you to transfer files securely between ECMWF and remote sites. Like the UNIX “`rcp`” command, `ectrans` requires no password to be specified for the remote host: the ECaccess gateway performs the security checking. Unlike standard FTP, `ectrans` is suitable for unattended file transfers in scripts, cron jobs, etc., as it avoids the problems inherent in storing passwords in text files and sending passwords across networks.

Even if you don't have a local gateway installed, you can benefit from the `ectrans` command by using the ECMWF ECaccess gateway. Please note that in this case the transfer is not as secured as when a Member State ECaccess gateway is used.

4.1 Target location

Users who wish to transfer files between ECMWF and Member State servers using `ectrans` need to declare one or more `ectrans` associations for the storage/retrieval of the remote file. This can be done either through the Web Toolkit command `ecaccess-association-put` (see section 5.8) or through the ECaccess Web interface of the target gateway (see section 7.4). For every association (previously known as “`msuser`”), the hostname, login username and password, target path and transfer protocol need to be specified.

Target directories can be located on:

- Member State servers running a standard FTP/SFTP service accessible from the ECaccess gateway. This is known as a “`genericFtp`”/“`genericSftp`” protocol and is the most convenient way of getting the files to the system you want, under the specified user ID.
- The server running the ECaccess gateway. This is known as a “`genericFile`” protocol. All users will share in a common directory the files transferred using this protocol.
- Member State servers running a proprietary application. The administrator provides `ectrans` with the implementation of the access protocol. The administrator can also use more complex rules to define special target locations for ECMWF users, Member State users or groups of Member State users.

4.2 ectrans command

With the `ectrans` command, Member State users can initiate secure file transfers between ECMWF (`ecgate` or HPCF systems) and a Member State server, as defined in the `ectrans` association.

When `ectrans` is used to put a file (from ECMWF to a remote server), the ECaccess Server will spool the file in the user's “`ectrans`” transfer queue. If the connection between the ECMWF and the remote gateway is down or if any error occurs, the file will be kept in the spool area at ECMWF and you can resume the transfer either through the web interface or with the Web Toolkit command `ecaccess-ectrans-restart`.

When `ectrans` is used to get a file (from a remote server to ECMWF) the transfer will fail by default, if the connection between the ECMWF and the remote gateway is down. A retry mechanism is available for all types of transfers. To show the `ectrans` usage:

```
> ectrans -help
usage: ectrans [-gateway name] -remote association@[protocol] \
  [-get|-put] -source [ec:|ectmp:]filename [args ...] (*)
ectrans -check requestID (*)
```

```

-gateway {arg} - access gateway name (default (**): eaccess.ecmwf.int)
-remote {arg} - association and (optional) protocol (default (**): *none*)
-source {arg} - source file name
-target {arg} - target file name (default: same as -source)
-mailto {arg} - target email address (default: current user)
-lifetime {arg} - lifetime of the file in the spool (default: 1w) (***) (****)
-delay {arg} - transmission delay (default: immediate transfer) (***) (****)
-at {arg} - transmission date (default: immediate transfer) (****)
-format {arg} - define the date format as used with -at (default: yyyyMMddHHmmss)
-retryCnt {arg} - define the number of retries (default: async=144, sync=0)
-retryFrq {arg} - define the frequency of retries (default: async=10m, sync=1m) (***)
-maxTime {arg} - define the maximum transfer duration (default: 12h) (***)
-priority {arg} - transmission priority 0-99 (default: 99) (****)
-put - interactive/synchronous transfer (no spool)
-get - interactive/synchronous pull (rather than push) file
-onsuccess - mail sent on successful transfer
-onfailure - mail sent when transfer has failed
-onretry - mail sent when transfer is retried
-keep - keep the request in the spool till expiration (****) (*****)
-remove - always remove the request from the spool (****) (*****)
-reject - if existing target file (default)
-append - if existing target file
-resume - if existing target file
-overwrite - if existing target file
-verbose - verbose mode on
-version - print version number
-help - this message

```

(*) If successful, a requestID is returned, which can be used in check requests. Exit code is 0 on success and >0 otherwise.

(**) The default values depend on the GATEWAY or REMOTE environment variables.

(***) Duration in weeks, days, hours, minutes or seconds (e.g. 1w|2d).

(****) These options are only relevant when the spool is used. The spool is no used during interactive transfers (-get and -put options).

(*****) By default, successful requests are removed from the spool and failed requests are kept in the spool till expiration.

The “reject”, “append”, “resume” and “overwrite” options are mutually exclusive and determine what to do if there is an existing target file. The “mailto” option specifies an email address to be notified in case of a successful (option “onsuccess”) and/or a failed transfer (option “onfailure”). The “check” option prints the status of the specified request on the standard output.

The transfer status, which can be checked with the `eaccess-ectrans-list` command or the Web interface, can takes values as listed in table 1.

Status	Meaning
INIT	Files are being transferred to the spool
COPY	Files are being transferred to the remote site
WAIT	Files are scheduled and waiting to be started
RETR	File transfer will be retried
STOP	Files have NOT been successfully transferred (error)
DONE	Files have been successfully transferred

Table 1: Transfer status.

4.2.1 Transfer to a Member State host via gateway

To transfer file “fff” from the current working directory on “ecgate” to the “genericFtp” protocol of the use “myUser” on the EAccess gateway “ecaccess.meteo.ms”:

```
> ectrans -gateway ecaccess.meteo.ms \  
          -remote myUser@genericFtp \  
          -source fff \  
          -verbose  
verbose: gateway=ecaccess.meteo.ms  
verbose: echost=ecgate.ecmwf.int  
verbose: ecport=644  
verbose: action=spool  
verbose: ecuser=uid  
verbose: source=fff  
verbose: target=fff  
verbose: keep=false  
verbose: remove=false  
verbose: option=reject  
verbose: lifetime=1w  
verbose: delay=(none)  
verbose: at=(now)  
verbose: format=yyyyMMddHHmmss  
verbose: retryCnt=144  
verbose: retryFrq=10m  
File to upload (5140480 bytes)  
9442903031
```

When a request has been spooled successfully, a requestID is returned immediately. `ectrans` will then return the exit code 0. The requestID can be used to reference the transfer, using the interface described in section 8 or with the command `ecaccess-ectrans-list`.

If the file is not successfully spooled, an error message is printed and the `ectrans` return code is -1.

4.2.2 Transfer from a Member State host via gateway

To transfer file “fff” at the “genericFtp” protocol of the “myUser” association of the EAccess gateway “ecaccess.meteo.ms” to the current directory at ECMWF:

```
> ectrans gateway ecaccess.meteo.ms \  
          -remote myUser@genericFtp \  
          -get -source fff \  
          -verbose  
gateway: ecaccess.ecmwf.int  
echost: ecgate.ecmwf.int  
ecport: 644  
action: get  
ecuser: uid  
target: fff  
source: fff  
keep : false  
option: reject  
File to download (0 bytes)  
5140480 bytes to download
```

When the request has been carried out successfully, the result is returned immediately. Transfers from a Members State to ECMWF are not spooled; they are carried out synchronously. The `ectrans` return code is 0 if the file has been transferred successfully or -1 if the file has not been transferred successfully.

5 Web Toolkit - The full featured client

The Web Toolkit, previously known as ECaccess shell commands or ECtools, is a set of Perl scripts for the management of files, file transfers, jobs, ectrans associations and events at ECMWF. They can be run by any user and on any Member State host.

The Web Toolkit is using SOAP to access the ECaccess web server.

Running these commands requires a valid certificate. The command `ecaccess-certificate-create` will create a certificate in the user's home directory (`$HOME/.eccert.crt`) from an ECMWF user identifier and a passcode (generated by a security token).

You need to ensure the following environment parameters are set with the correct values:

```
http_ecaccess=gateway.meteo.ms:9080
https_ecaccess=gateway.meteo.ms:9443
```

(e.g. if your local ECaccess Gateway name is "gateway.meteo.ms" and you are using the default ECaccess http/s ports 9080/9443)

The default values are pointing to the `ecaccess.ecmwf.int` server.

Your gateway administrator can provide other default values for these parameters. However, your environment variables take precedence over these default values.

If the directory containing the Web Toolkit commands is not in your command path or you do not know the directory in which the commands are installed, try running the "ecaccess-gateway-name" command. If the command is not available, you will need to contact your Computing Representative, your local ECaccess administrator - if known - or User Support at ECMWF. Alternatively, you may wish to install the Web Toolkit yourself (see <http://www.ecmwf.int/services/ecaccess/download/>).

The ECaccess Tools are organized in sets covering access to the whole computing and archiving facilities of ECMWF and are described in the following sections. Each command is documented with its own man page which provide explanation as well as examples on how to use it.

The following options are common to all the Web Toolkit commands:

- help Print a brief help message and exits.
- manual Prints the manual page and exits.
- debug Display the SOAP messages exchanged.
- version Print the ECaccess version number.

5.1 Certificate management

The ECaccess certificate can be created using the command `ecaccess-certificate-create`. From an ECMWF user identifier and a PASSCODE (using a security token), it generates a certificate in `$HOME/.eccert.crt`.

Alternatively, a certificate can be created using the ECaccess Web interface, see section 7.

The Web Toolkit is also available at ECMWF. In contrast to using the Web Toolkit installed at your site you will not need a certificate when using it at ECMWF as you have already been validated at login.

To display a help screen describing the `ecaccess-certificate-create` usage:

```
> ecaccess-certificate-create -help
Usage:
    ecaccess-certificate-create [options] [user-id]
```

```
Options: -help brief help message -manual full documentation -debug
enable messages output
```

```
Options:
  -help      Print a brief help message and exits.

  -manual    Prints the manual page and exits.

  -debug     Display the SOAP messages exchanged.
```

Certificates are PEM/Base64 encoded ASCII files.

OpenSSL can be used to decode and display certificate components.

The expiry of the various EAccess services can be displayed with the command [ecaccess-certificate-list](#):

```
ecgate{/home/ectrain/trx}:1 --> ecaccess-certificate-list
submitJob          168h   Jan 18 11:56   job submission
getJobList         168h   Jan 18 11:56   job list
deleteJob          168h   Jan 18 11:56   delete a job
getJobResult       168h   Jan 18 11:56   job result
deleteFile         168h   Jan 18 11:56   delete file
getFileList        168h   Jan 18 11:56   get file list
mkdir              168h   Jan 18 11:56   make directory
getFileSize        168h   Jan 18 11:56   get file size
readFile           168h   Jan 18 11:56   read file
writeFile          168h   Jan 18 11:56   write file
moveFile           168h   Jan 18 11:56   move file
rmdir              168h   Jan 18 11:56   remove directory
chmod              168h   Jan 18 11:56   change file mode
getTempFile        168h   Jan 18 11:56   create temporary file
getTransferList    168h   Jan 18 11:56   get transfer list
```

As can be seen from the output, for a normal user-id the validity is 168 hours (7 days) for all services. The date/time shown refers to the expiration of the certificate.

5.2 General information

To display the "Message of the day" (/etc/motd on ecgate) containing announcements of upcoming system sessions etc., use the command [ecaccess-cosinfo](#):

```
-> ecaccess-cosinfo
*****
*
*
*   Welcome to AIX Version 5.3!
*
*
*
*   Please see the README file in /usr/lpp/bos for information pertinent to
*   this release of the AIX Operating System.
*
*
*****

ECMWF SYSTEM SESSIONS
-----

WEDNESDAY the 12th of JANUARY 2011:
```

```
-----
08:30-10:30 UTC Mars and ECFS System Session:
impact: Mars and ECFS will be unavailable

To reread the message please use: more /etc/motd or cat /etc/motd
=====
```

5.3 Gateway information

Commands named `ecaccess-gateway-*` (table 2) provide information about ECaccess gateways (see section 2.1).

Command	Purpose
<code>ecaccess-gateway-list</code>	List of ECaccess Gateways
<code>ecaccess-gateway-name</code>	Display the name of your default ECaccess Gateway

Table 2: Web Toolkit commands providing information on ECaccess gateways (`ecaccess-gateway-*`).

The command `ecaccess-gateway-name` is the only one which can be used without authentication/certificate.

5.4 File management

Files at ECMWF can be managed through the Web Toolkit commands named `ecaccess-file-*`. The file location is specified with the following syntax

```
[domain:][/user-id/]path
```

where `domain:` can take the values as listed in table 3. The `user-id` refers to a an ECMWF computer user-id. Only if the domain is specified as an ECFS domain, then the `user-id` could also be a common pool, e.g. `demeter`.

Domain value	Purpose
<code>home:</code>	the <code>ecgate \$HOME</code> file system
<code>scratch:</code>	the <code>ecgate \$SCRATCH</code> file system
<code>ec:</code>	the ECFS domain <code>ec:</code>
<code>ectmp:</code>	the ECFS domain <code>ectmp:</code>
<code>host-name:</code>	any server at ECMWF, e.g. <code>c2a</code>

Table 3: Domain values.

If no domain is specified then an absolute path will translate to an absolute path on the `ecgate` server and a relative path will translate to a path in the `HOME` directory of the current user.

If no `user-id` is specified then the current `user-id` is selected by default. When you specify a `host-name` you are by default under the root directory; the `user-id` parameter can not be used with a `host-name`.

File location examples:

bin/a.out	file a.out in directory \$HOME/bin of the current user
home:bin/a.out	file a.out in directory \$HOME/bin of the current user
/tmp/a.out	file a.out in directory /tmp on ecgate
home:/xzy/bin/a.out	file a.out in directory \$HOME/bin of user xzy
ec:bin/a.out	file a.out in directory bin in the current user's ECFS domain ec:
ec:/xzy/bin/a.out	file a.out in directory bin in user xzy's ECFS domain ec:
c2a:/c2a/tmp/group/xzy/a.out	file a.out in directory /c2a/tmp/systems/xzy/ on c2a

Command	Purpose
ecaccess-file-chmod	Change EAccess File Mode Bits
ecaccess-file-copy	Copy an EAccess File
ecaccess-file-delete	Remove an EAccess File
ecaccess-file-dir	List EAccess Directory Contents
ecaccess-file-get	Download an EAccess File
ecaccess-file-mget	Download Multiple EAccess Files at once
ecaccess-file-mkdir	Make a Directory on the EAccess File System
ecaccess-file-modtime	Show the Last Modification Time of an EAccess File
ecaccess-file-move	Move or Rename EAccess Files
ecaccess-file-mput	Upload Multiple Local Files on the EAccess File System at once
ecaccess-file-put	Upload a File on the EAccess File System
ecaccess-file-rmdir	Remove a Directory on the EAccess File System
ecaccess-file-size	Show the Size of an EAccess File

Table 4: Web Toolkit commands for file management (*ecaccess-file-**).

5.5 Batch job management

Batch jobs at ECMWF can be managed through the Web Toolkit commands named *ecaccess-job/queue-**, see table 5. Possible values of the job status, which can be checked with the command *ecaccess-job-list* or via the Web interface, are listed in table 6.

A special service (see option *-eventIds*) allows to automatically submit jobs to be run when certain points in the daily ECMWF operational forecast suite have been reached. The main purpose is to ensure that certain data are available before e.g. submitting a MARS request. These events correspond to the different stages when the ECMWF operational activity has produced certain data or products. The list of events can be retrieved with the “*ecaccess-event-list*” command.

Command	Purpose
ecaccess-job-delete	Delete an EAccess Job
ecaccess-job-get	Download a Job Output/Input/Error File
ecaccess-job-list	List all EAccess Jobs
ecaccess-job-restart	Restart an EAccess Job
ecaccess-job-submit	Submit a new EAccess Job
ecaccess-queue-list	List available queues

Table 5: Web Toolkit commands for batch job management (*ecaccess-job/queue-**).

Status	Meaning
DONE	Jobs have successfully completed
EXEC	Jobs are running
INIT	Jobs are being initialised
RETR	Jobs will be resubmitted
STDBY	Jobs are waiting for an event
STOP	Jobs have NOT completed (error)
WAIT	Jobs have been queued to the scheduler (e.g. LoadLeveler)

Table 6: Job status.

Command	Purpose
ecaccess-event-clear	Clear an ECaccess Event
ecaccess-event-create	Create an ECaccess Event
ecaccess-event-delete	Delete an ECaccess Event
ecaccess-event-grant	Grant usage of an ECaccess Event
ecaccess-event-list	List available events
ecaccess-event-send	Trigger an ECaccess Event

Table 7: Web Toolkit commands for management of events at ECMWF (*ecaccess-event-**).

5.6 Management of events

ECMWF maintains some notifications (events) which are linked to ECMWF's operational activity and offers the service for time-critical jobs (see also separate documentation at http://www.ecmwf.int/services/computing/docs/tc_apps/index.html). This service is also available to MS users who maintain their own notifications and can therefore create simple dependencies between different activities, at ECMWF and remote sites.

The shell commands to managed events are listed in table 7.

5.7 Management of Ectrans transfers

The commands for the management of ECMWF-initiated transfers (ectrans, see section 4.2) are listed in table 8. They can only be used for the management of transfers, which have used the ECaccess gateway as shown with the `ecaccess-gateway-name` command.

Command	Purpose
ecaccess-ectrans-delete	Delete Ectrans
ecaccess-ectrans-list	List all ectrans transfers
ecaccess-ectrans-request	Request a new Ectrans transfer
ecaccess-ectrans-restart	Restart an existing Ectrans transfer

Table 8: Web Toolkit commands for management of ECMWF-initiated transfers (*ecaccess-ectrans-**).

5.8 Management of Ectrans associations

Before making use of ectrans, users will need to declare an ectrans association (previously referred to as 'remote Member State user (msuser)') for the storage/retrieval of the remote file. The management of these

associations can be done through the Web Toolkit commands named `ecaccess-association-*` listed in table 9.

Command	Purpose
ecaccess-association-delete	Delete Association
ecaccess-association-get	Get the Association Descriptive File
ecaccess-association-list	List your ECTrans associations
ecaccess-association-protocol	List the supported ECTrans Protocol
ecaccess-association-put	Update/Create an Association

Table 9: Web Toolkit commands for management of ECTrans associations (`ecaccess-association-*`).

5.9 Execution return codes

The option `-debug` can be used with any Web Toolkit command to display information concerning the SOAP messages exchanged. If a Web Toolkit command doesn't work correctly, please run it with the `-debug` option and send the output to ECMWF for further investigation.

Web Toolkit commands return 0 if successful, and a non-zero code otherwise. Each time an error occurs, a message indicating the error is displayed to the user.

6 Web Services - Standard thin clients

If you are NOT using the ECMWF Gateways (`ecaccess` or `msaccess`) then please refer to the [previous user guide](#) for accessing the ECaccess Web Services V1. This documentation is about ECaccess Web Services V2 which is currently only available at ECMWF.

At heart, ECaccess Web Services APIs use JSON as a data format and HTTPS as a protocol for data transmission. ECMWF has released a full-featured Perl client library to make interacting with ECaccess Web Services APIs easier. However, if Perl is not available on your platform or you are not able to install additional software then CURL is an alternative to interact with the ECaccess Web Services.

CURL is a command-line application for performing requests using a variety of protocols including HTTPS, required to interact with the APIs at a low level.

In this document we assume that you are already familiar with the ECaccess Services. If this is not the case then please have refer to other sections in this [User Guide](#) to learn about the concepts behind Gateways, Operations, Associations, Protocols, Files, Jobs, Events, Queues and Transfers.

6.1 Obtaining CURL

CURL is commonly available on a default install of many UNIX/Linux platforms. Try typing `curl` in your favorite shell to see if it is installed and in your PATH. If you don't have the tool installed, visit the download page on the CURL website to obtain the official source or a user-contributed binary package:

<http://curl.haxx.se/download.html>

Note that the command-line tool uses the `libcurl` library, which may be offered as a separate download package, so, if you're not compiling from the source, be sure to download a *binary* package instead of a *libcurl* package. The SSL-enabled packages are required to use the ECaccess Web Services.

6.2 Environment

You need to ensure the following environment variables are set with the correct values:

```
> export ECACCESS_URLV2="https://ecaccess.ecmwf.int/ecaccess/v2"  
> export ECACCESS_OPTS="-w \ncode=%{http_code}\n -3 -k"
```

If you are using the RMDCN network to access ECMWF then please use `msaccess.ecmwf.int` instead of `ecaccess.ecmwf.int`.

The curl options used are the following:

- w display the http code on stdout (`\ncode=%{http_code}\n`)
- 3 forces curl to use SSL version 3
- k disable certificate checking

If you don't want or can't setup these environment parameters then just replace `$ECACCESS_URLV2` and `$ECACCESS_OPTS` by their contents in the rest of the document.

6.3 Input/output

For each ECAccess Service available an example is given in the rest of this document with the CURL syntax to use, the list of parameters and the expected output.

Most of the services will return a JSON message. JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

Each JSON message should include a field "success" set to "yes". If an error occurs then this field is set to "no" and an "error" field gives the error message returned by the service.

For more informations concerning JSON see <http://www.json.org/>

There are a few occasions when an error will not be indicated via a JSON message but with an http error code (e.g. when a binary stream is expected rather than a message):

401 Unauthorized	authentication error when trying to download a new certificate
400 Bad request	error when downloading a data file or job output/input file.
403 Forbidden	access is only allowed through a secure connection (https).
412 Precondition failed	missing mandatory URL parameter or unknown JSON parameter.
404 Not Found	the requested service does not exist.

On a successful request the http status code returned by the server is the following:

200 OK the request has proceeded a JSON message.

If you get an HTML document then that could be a problem with a proxy. In this case the HTML message should give you an indication of what the problem is.

6.4 Authentication

6.4.1 Certificates

Accessing the ECAccess Web Services API requires a valid certificate. Certificates can be created with the CURL command

```
> curl $EACCESS_OPTS -c $HOME/.eccert.txt \  
"$EACCESS_URLV2/cookie?ecuser=uid&passcode=XXXXXXXX"
```

which (if successful) returns

```
{  
  "success" : "yes"  
}  
code=200
```

In this request just replace uid by your ECMWF identifier and XXXXXXXX by your PASSCODE obtained from your security token. It generates a certificate in ".eccert.txt" in your user's home directory.

The EAccess Certificate is valid for one week and should be renewed after expiration.

6.4.2 Tokens

Requests to the EAccess Web Services are performed by adding an HTTP header to the request which contains an Authentication token. Authentication tokens are generated for each session and require a valid certificate.

To get an Authentication token (e.g. LOGIN) use the command

```
> curl $EACCESS_OPTS -b $HOME/.eccert.txt "$EACCESS_URLV2/cookie/token"
```

which returns

```
{  
  "token" : "ahs4w294",  
  "success" : "yes"  
}  
code=200
```

Your new token is `ahs4w294`. You can now use this token for several requests to the EAccess Web Services. Once you have finished with your token and don't plan to use it any more then you can release it (e.g. LOGOUT) with

```
> curl $EACCESS_OPTS -X DELETE -H "Token: ahs4w294" "$EACCESS_URLV2/token"
```

which returns

```
{  
  "success" : "yes"  
}  
code=200
```

The Authentication token is now released and can't be used for any subsequent requests.

6.5 Cosinfo

Cosinfo provides ECMWF computer operations system information, such as scheduled or unscheduled down-time of servers, applications or hardware. The command

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/cosinfo"
```

returns e.g.

```
{
  "cosinfo" :
  "*****\n*
*\n*
*\n* Welcome to AIX Version 5.3!
*\n*
*\n*
*\n* Please see the README file in /usr/lpp/bos for information pertinent to
*\n* this release of the AIX Operating System.
*\n*
*\n*
*\n*****\n\n\nDear
colleagues,\n\n\nThe following system sessions have been announced for\n\n
Wednesday 5th of October 2012\n\n\n08:00-10:00 UTC   DHS system
session\nImpact:   MARS and ECFS may be affected\n\n\n13:00-14:30 UTC
ECPDS system session\nImpact:   none\n\n\nHave a nice day,\nSylvia for
Calldesk\n\n\n\nTo reread the message please use: more /etc/motd or cat
/etc/motd\n\n=====\n",
  "success" : "yes"
}
code=200
```

6.6 Gateways

To display the name of the Gateway you are connected to (please note this command does not need a token), command

```
> curl $ECACCESS_OPTS "$ECACCESS_URLV2/gateway/name"
```

returns

```
{
  "gateway" : "ecaccess.ecmwf.int",
  "success" : "yes"
}
code=200
```

To display the connection status of the ECaccess Gateway you are connected to (please note this command does not need a token),

```
> curl $ECACCESS_OPTS "$ECACCESS_URLV2/gateway/connected"
```

returns

```
{
  "connected" : true,
  "success" : "yes"
}
code=200
```

All ECaccess Gateways can be listed with

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/gateways"
```

which returns

```
{
  "gateways" : [ {
    "name" : "forte.meteo.fr",
    "comment" : "HP-UX B.11.00 - Java HotSpot(TM) Server VM 1.4.2
1.4.2.08-050401-19:15-PA_RISC1.1 PA1.1 mixed mode Hewlett-Packard
Company",
    "version" : "3.1.0_2007011001",
    "active" : false,
    "lastReport" : "Aug 22 13:15"
  }, {
    "name" : "ecaccess.knmi.nl",
    "comment" : "Linux 2.6.18-92.1.13.el5 - Java HotSpot(TM) Server VM
11.0-b16 mixed mode Sun Microsystems Inc.",
    "version" : "3.3.0_2009021801",
    "active" : true,
    "lastReport" : "Sep 27 15:46"
  }
  (...
  ] ],
  "success" : "yes"
}
code=200
```

When a gateway name is specified in the URL then the details for this Gateway only are displayed:

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/gateways/forte.meteo.fr"
```

returns

```
{
  "gateway" : {
    "name" : "forte.meteo.fr",
    "comment" : "HP-UX B.11.00 - Java HotSpot(TM) Server VM 1.4.2
1.4.2.08-050401-19:15-PA_RISC1.1 PA1.1 mixed mode Hewlett-Packard
Company",
    "version" : "3.1.0_2007011001",
    "active" : false,
    "lastReport" : "Aug 22 13:15"
  },
  "success" : "yes"
}
code=200
```

6.7 Operations

To list all the ECaccess Operations/Services available with the current token use

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/operations"
```

which returns

```
{
  "operations" : [ {
```

```

    "name" : "submitJob",
    "comment" : "job submission",
    "duration" : "168h",
    "startDate" : "Sep 30 11:40",
    "validity" : 7,
    "endDate" : "Oct 07 11:40",
    "standard" : "168h"
  }, {
    "name" : "getJobList",
    "comment" : "job list",
    "duration" : "168h",
    "startDate" : "Sep 30 11:40",
    "validity" : 7,
    "endDate" : "Oct 07 11:40",
    "standard" : "168h"
  }
  (... )
} ],
"success" : "yes"
}
code=200

```

If you specify an operation name in the URL then the information for this operation only will be displayed, e.g.

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/operations/submitJob"
```

returns

```

{
  "operation" : {
    "name" : "submitJob",
    "comment" : "job submission",
    "duration" : "168h",
    "startDate" : "Sep 30 11:40",
    "validity" : 7,
    "endDate" : "Oct 07 11:40",
    "standard" : "168h"
  },
  "success" : "yes"
}
code=200

```

6.8 Associations

All your ECTrans Associations for a Gateway (e.g. the “ecaccess.ecmwf.int” Gateway) can be listed with

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" \
  "$ECACCESS_URLV2/associations/ecaccess.ecmwf.int"
```

which returns

```

{
  "associations" : [ {
    "name" : "test2",
    "protocol" : "",
    "hostName" : "193.40.11.180",
    "owner" : "uid",

```

```
    "comment" : "test association",
    "active" : false,
    "data" : "...",
    "login" : "test",
    "directory" : "test",
    "grantedUserList" : ""
    (...)
  } ],
  "success" : "yes"
}
```

code=200

When an association name is specified in the URL then the details for this Association only are displayed (e.g. test2):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" \
  "$ECACCESS_URLV2/associations/ecaccess.ecmwf.int/test2"
```

returns

```
{
  "association" : {
    "name" : "test2",
    "protocol" : "",
    "hostName" : "193.40.11.180",
    "owner" : "uid",
    "comment" : "test association",
    "active" : false,
    "data" : "...",
    "login" : "test",
    "directory" : "test",
    "grantedUserList" : ""
  },
  "success" : "yes"
}
```

code=200

To update or create an ECTrans Association (e.g. if the name test3 exists then the Association is updated otherwise it is created) use

```
> curl $ECACCESS_OPTS -X PUT -H "Token: ahs4w294" -H "Content-Type: application/json" \
-d '{ \
  "name":"test3", \
  "protocol":"genericFtp", \
  "hostName":"ecaccess", \
  "comment":"test association", \
  "active":true, \
  "login":"uid", \
  "password" : "*****" \
}' $ECACCESS_URLV2/associations/ecaccess.ecmwf.int
```

which returns

```
{
  "success" : "yes"
}
```

code=200

To delete an ECTrans Association (once deleted the Association can not be retrieved and its associated History is removed from the ECAccess Gateway Database) use

```
> curl $ECACCESS_OPTS -X DELETE -H "Token: ahs4w294" \
  "$ECACCESS_URLV2/associations/ecaccess.ecmwf.int/test3"
```

which returns

```
{
  "success" : "yes"
}
code=200
```

All Parameters for the ECTrans Association are listed in table 10.

Type	Name	Comment
String	name	
Boolean	active	default FALSE
String	comment	
String	grantedUserList	multiple users separated by a ','
String	data	
String	directory	
String	hostName	
String	login	
String	password	
String	protocol	

Table 10: Parameters for the ECTrans Association. For additional information please see the [ecaccess-association-put manpage](#).

6.9 Protocols

To list all the ECTrans Protocols supported by a Gateway (e.g. ecaccess.ecmwf.int) command

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/protocols/ecaccess.ecmwf.int"
```

returns

```
{
  "protocols" : [ {
    "name" : "genericFile",
    "value" :
      "${ecmwf.dir}/gateway/ectrans/$msuser[name]/$msuser[dir]$target",
    "comment" : "Save files on the gateway",
    "active" : true,
    "archive" : "${ecmwf.dir}/gateway/lib/ectrans",
    "module" : "file",
    "activeModule" : true
    (...)
  } ],
  "success" : "yes"
}
code=200
```

When a protocol-name is specified in the URL then the details for this Protocol only are displayed (.e.g. genericFtp):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" \  
"$ECACCESS_URLV2/protocols/ecaccess.ecmwf.int/genericFtp"
```

returns

```
{  
  "protocol" : {  
    "name" : "genericFtp",  
    "value" :  
    "$msuser[login]:$msuser[passwd]@$msuser[host]/$msuser[dir]$target",  
    "comment" : "Generic FTP access from the gateway",  
    "active" : true,  
    "archive" : "${ecmwf.dir}/gateway/lib/ectrans",  
    "module" : "ftp",  
    "activeModule" : true  
  },  
  "success" : "yes"  
}  
code=200
```

6.10 Files

The ECaccess files are in the form [domain:][user-id/]path. Please refer to section 5.4 for more information on the ECaccess File System.

To download a source ecaccess file, e.g. c2a:/tmp/test.tar.gz), use

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" -o test.tar.gz \  
"$ECACCESS_URLV2/file?source=c2a:/tmp/test.tar.gz&offset=0"
```

The file will be downloaded as ./test.tar.gz

To upload a file (e.g. test.tar.gz):

```
> curl $ECACCESS_OPTS -X POST -H "Token: ahs4w294" \  
-T test.tar.gz "$ECACCESS_URLV2/file?target=test.tar.gz&offset=0&umask=777"
```

The file will be uploaded as test.tar.gz on ecgate.

To delete an ecaccess-file (.e.g. test.tar.gz) use

```
> curl $ECACCESS_OPTS -X DELETE -H "Token: ahs4w294" \  
"$ECACCESS_URLV2/file?source=test.tar.gz"
```

returns

```
{  
  "success" : "yes"  
}  
code=200
```

To change the file mode of an ecaccess-file, .e.g. set the mode of test.tar.gz to 640,

```
> curl $ECACCESS_OPTS -X PUT -H "Token: ahs4w294" \  
"$ECACCESS_URLV2/chmod?source=test.tar.gz&mode=640"
```

returns

```
{  
  "success" : "yes"  
}  
code=200
```

To create a new directory (e.g. c2a:/tmp/TESTDIR):

```
> curl $ECACCESS_OPTS -X PUT -H "Token: ahs4w294" \  
"$ECACCESS_URLV2/mkdir?directory=c2a:/tmp/TESTDIR"
```

returns

```
{  
  "success" : "yes"  
}  
code=200
```

To delete a directory (e.g. c2a:/tmp/TESTDIR):

```
> curl $ECACCESS_OPTS -X DELETE -H "Token: ahs4w294" \  
"$ECACCESS_URLV2/rmdir?directory=c2a:/tmp/TESTDIR"
```

returns

```
{  
  "success" : "yes"  
}  
code=200
```

To get the last modified date and time of an ecaccess-file (e.g. README):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" \  
"$ECACCESS_URLV2/mdtm?source=README"
```

returns

```
{  
  "mdtm" : "Sep 14 14:14",  
  "success" : "yes"  
}  
code=200
```

To get the size of an ecaccess-file (e.g. README):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" \  
"$ECACCESS_URLV2/size?source=README"
```

returns

```
{
  "success" : "yes",
  "size" : 5205
}
code=200
```

To copy an eaccess-file to another eaccess-file. In this case the original eaccess-file is not deleted (e.g. README -j README2)

```
> curl $ECACCESS_OPTS -X PUT -H "Token: ahs4w294" \
  "$ECACCESS_URLV2/copy?source=README&target=README2&erase=false"
```

returns

```
{
  "success" : "yes"
}
code=200
```

Copy and delete the original eaccess-file (e.g. README-jREADME2):

```
> curl $ECACCESS_OPTS -X PUT -H "Token: ahs4w294" \
  "$ECACCESS_URLV2/copy?source=README&target=README2&erase=true"
```

returns

```
{
  "success" : "yes"
}
code=200
```

Move an eaccess-file (e.g. README -j README2)

```
> curl $ECACCESS_OPTS -X POST -H "Token: ahs4w294" \
  "$ECACCESS_URLV2/move?source=README&target=README2"
```

returns

```
{
  "success" : "yes"
}
code=200
```

Get a list of files and directories (e.g. in scratch:dir):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" \
  "$ECACCESS_URLV2/dir?path=scratch:dir"
```

returns

```
{
  "dir" : [ {
    "name" : "Le Sapin de Noel.doc",
    "permissions" : "-rw-r--r--",
    "size" : "24064",
    "path" : ""
  }
]
```

```

    "comment" : "",
    "time" : "Dec 14 2009",
    "domain" : "scratch:",
    "user" : "uid",
    "dir" : false,
    "group" : "systems",
    "link" : ""
  (...)
} ],
"success" : "yes"
}
code=200

```

Create a temporary file:

```
> curl $ECACCESS_OPTS -X PUT -H "Token: ahs4w294" \
  "$ECACCESS_URLV2/mktmp"
```

returns

```

{
  "mktmp" : "tmp:roqZya",
  "success" : "yes"
}
code=200

```

This temporary file "tmp:roqZya" can be used to upload a temporary file (e.g. when submitting the input file for an ECaccess Job).

6.11 Jobs

To submit a new ECaccess Job, e.g. myjob.ksh which is a shell script without Loadleveler directives located in the ECMWF home directory:

```
> curl $ECACCESS_OPTS -X POST -H "Token: ahs4w294" \
  -H "Content-Type: application/json" \
  -d '{ \
    "queueName": "ecgate", \
    "inputFile": "home:myjob.ksh", \
    "containsDirectives": false \
  }' \
  "$ECACCESS_URLV2/jobs"
```

returns

```

{
  "jobId" : "2368129",
  "success" : "yes"
}
code=200

```

All Job Request parameters are listed in table 11.

To list all your ECaccess Jobs:

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/jobs"
```

Type	Name	Comment
String	scheduledDate	in the form 'yyyy-MM-dd HH:mm'
String	userMailAddress	
Boolean	sendMailOnStart	
Boolean	sendMailOnSuccess	
Boolean	sendMailOnFailure	
Boolean	sendMailOnRetry	
Boolean	containsDirectives	default TRUE
String	queueName	default is ecgate
String	name	
String	transferGatewayName	default is current Gateway
String	transferRemoteLocation	in the format association-name[@protocol]
String	transferOutputFile	
String	transferErrorFile	
String	transferInputFile	
Boolean	transferKeepInSpool	default FALSE
String	eventIds	list separated by ';' or ','
Boolean	renewSubscription	default FALSE
Boolean	errorToOutput	default FALSE
String	manPageContent	
Integer	lifeTime	in days
Integer	retryCount	default 0
Integer	retryFrequency	in seconds
String	inputFile	file on ECMWF file system

Table 11: Parameters for a Job Request. For additional information please see the [ecaccess-job-submit manpage](#).

returns

```
{
  "jobs" : [ {
    "name" : "",
    "hostName" : "ecgate.ecmwf.int",
    "comment" : "",
    "status" : "DONE",
    "jobId" : "2368129",
    "tryCount" : 1,
    "tryDone" : 1,
    "scheduledDate" : "Sep 30 13:51",
    "eventIds" : [ ],
    "queueName" : "ecgate",
    "errorFileSize" : 122,
    "expirationDate" : "-",
    "inputFileSize" : 43,
    "outputFileSize" : 1451,
    "schedulerJobId" : "ecga00.ecmwf.int.1956629",
    "schedulerName" : "LoadLeveler"
    (...)
  } ],
  "success" : "yes"
}
code=200
```

To get an ECaccess Job Output (e.g. 2368129):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/jobs/2368129/output"
```

To get an ECaccess Job Error (e.g. 2368129):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/jobs/2368129/error"
```

To get an ECaccess Job Input (e.g. 2368129):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/jobs/2368129/input"
```

To get the details for a specific ECaccess Job (e.g. 2368129) just add the ECaccess Job Identifier in the URL:

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/jobs/2368129"
```

returns

```
{
  "job" : {
    "name" : "",
    "hostName" : "ecgate.ecmwf.int",
    "comment" : "",
    "status" : "DONE",
    "jobId" : "2368129",
    "tryCount" : 1,
    "tryDone" : 1,
    "scheduledDate" : "Sep 30 13:51",
    "eventIds" : [ ],
    "queueName" : "ecgate",
    "errorFileSize" : 122,
    "expirationDate" : "Oct 07 13:51",
    "inputFileSize" : 43,
    "outputFileSize" : 1451,
    "schedulerJobId" : "ecga00.ecmwf.int.1956629",
    "schedulerName" : "LoadLeveler"
  },
  "success" : "yes"
}
code=200
```

To restart an existing ECaccess Job (e.g. 2368129):

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/jobs/2368129"
```

returns

```
{
  "success" : "yes"
}
code=200
```

To delete an ECaccess Job (e.g. 2368129):

```
> curl $ECACCESS_OPTS -X DELETE -H "Token: ahs4w294" "$ECACCESS_URLV2/jobs/2368129"
```

returns

```
{
  "success" : "yes"
}
code=200
```

6.12 Events

To list all the EAccess Events which are available to your ECMWF user identifier

```
> curl $EACCESS_OPTS -H "Token: ahs4w294" "$EACCESS_URLV2/events"
```

returns

```
{
  "events" : [ {
    "name" : "an00h000",
    "comment" : "At this stage, the analysis at 00UTC is complete.",
    "metadata" : "",
    "eventId" : "167",
    "title" : "",
    "isPublic" : true,
    "ownerUserId" : "emos",
    "notifyUserList" : "",
    "subscribeUserList" : ""
    (...)
  } ],
  "success" : "yes"
}
code=200
```

To get the details for a specific event, e.g. an00h000, just add the event name or identifier (eventId) in the URL, e.g.

```
> curl $EACCESS_OPTS -H "Token: ahs4w294" "$EACCESS_URLV2/events/an00h000"
```

returns

```
{
  "event" : {
    "name" : "an00h000",
    "comment" : "At this stage, the analysis at 00UTC is complete.",
    "metadata" : "",
    "eventId" : "167",
    "title" : "",
    "isPublic" : true,
    "ownerUserId" : "emos",
    "notifyUserList" : "",
    "subscribeUserList" : ""
  },
  "success" : "yes"
}
code=200
```

To create an EAccess Event:


```
> curl $ECACCESS_OPTS -X POST -H "Token: ahs4w294" -H "Content-Type: application/json" \
-d '{ \
  "name":"myEvent", \
  "comment":"This is for test", \
  "isPublic":false}' \
"$ECACCESS_URLV2/events"
```

returns

```
{
  "eventId" : "807",
  "success" : "yes"
}
code=200
```

All parameters for the Event Request are listed in table 12.

Type	Name	Comment
String	name	
String	comment	
Boolean	isPublic	default FALSE
Boolean	overwrite	default FALSE
String	title	
String	metadata	

Table 12: Parameters for an Event Request. For additional information please see the [ecaccess-event-create manpage](#).

To grant an ECaccess Event (e.g. 807):

```
> curl $ECACCESS_OPTS -X POST -H "Token: ahs4w294" -H "Content-Type: application/json" \
-d '{"subscribe":true,"notify":true,"userList":"uid,abc"}' \
"$ECACCESS_URLV2/events/807/grant"
```

returns

```
{
  "success" : "yes"
}
code=200
```

All parameters for the Grant Event Request are listed in table 13.

Type	Name	Comment
Boolean	subscribe	default FALSE
Boolean	notify	default FALSE
String	userList	multiple users separated by a ','

Table 13: Parameters for a Grant Event Request. For additional information please see the [ecaccess-event-grant manpage](#).

To trigger an ECaccess Event (e.g. 807):

```
> curl $ECACCESS_OPTS -X POST -H "Token: ahs4w294" -H "Content-Type: application/json" \
-d '{"env":"TEST=TEST","seq":"1"}' "$ECACCESS_URLV2/events/807/send"
```

returns

```
{
  "count" : 0,
  "success" : "yes"
}
code=200
```

All parameters for the Send Event Request are listed in table 14.

Type	Name	Comment
String	env	multiple variables separated by a ';'
String	delay	specified in [w]eeks, [d]ays, [h]ours, [m]inutes or [s]econds (e.g. 1w or 2d)
String	at	in the format 'yyyy-MM-dd HH:mm'
Integer	seq	

Table 14: Parameters for the Send Event Request. For additional information please see the [ecaccess-event-send manpage](#).

To clear an ECaccess Event (e.g. 807):

```
> curl $ECACCESS_OPTS -X PUT -H "Token: ahs4w294" "$ECACCESS_URLV2/events/807/clear"
```

returns

```
{
  "success" : "yes"
}
code=200
```

To delete an ECaccess Event (e.g. 807):

```
> curl $ECACCESS_OPTS -X DELETE -H "Token: ahs4w294" "$ECACCESS_URLV2/events/807"
```

returns

```
{
  "success" : "yes"
}
code=200
```

6.13 Queues

To list the available ECaccess Queues:

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/queues"
```

returns

```
{
  "queues" : [ {
    "hostName" : "ecgate.ecmwf.int",
    "comment" : "submission on ecgate",

```

```

    "active" : true,
    "queueName" : "ecgate",
    "schedulerName" : "LoadLeveler",
    "numberOfJobsInDoneState" : 7203,
    "numberOfJobsInExecState" : 13,
    "numberOfJobsInInitState" : 1275,
    "numberOfJobsInStopState" : 84,
    "numberOfJobsInWaitState" : 1
    (...)
  } ],
  "success" : "yes"
}
code=200

```

To get the details for a specific EAccess Queue (e.g. ecgate) just add the queue name in the URL:

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/queues/ecgate"
```

returns

```

{
  "queue" : [ {
    "name" : "pvwave",
    "comment" : "For jobs using PV-Wave - restricted access"
  }, {
    "name" : "timecrit",
    "comment" : "Suitable for time critical jobs - restricted access"
  } ],
  (...)
} ],
"success" : "yes"
}
code=200

```

6.14 Transfers

In order to list all your ETrans Transfers in the ETrans Spool:

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/transfers"
```

returns

```

{
  "transfers" : [ ],
  "success" : "yes"
}
code=200

```

To get the details for a specific ETrans Transfer (e.g. 32595048) just add the Transfer Identifier in the URL:

```
> curl $ECACCESS_OPTS -H "Token: ahs4w294" "$ECACCESS_URLV2/transfers/32595048"
```

returns

```

{
  "transfer" : {
    "hostName" : "ecaccess.ecmwf.int",

```

```
    "comment" : "",
    "sourceFileName" : "/home/group/uid/ecmwf.properties",
    "status" : "STOP",
    "keepInSpool" : true,
    "fileSize" : 5205,
    "tryCount" : 145,
    "tryDone" : 1,
    "scheduledDate" : "Sep 30 14:26",
    "targetFileName" : "ecmwf.properties",
    "ownerUserId" : "test",
    "lastErrorMessage" : "user not granted by uid for test",
    "transferId" : "32595048"
  },
  "success" : "yes"
}
code=200
```

In order to request a new ECTrans transfer:

```
> curl $ECACCESS_OPTS -X POST -H "Token: ahs4w294" -H "Content-Type: application/json" \
-d '{ \
  "remoteLocation":"test", \
  "sourceFileName":"ecmwf.properties", \
  "gatewayName":"ecaccess.ecmwf.int", \
  "keepInSpool":true}' \
"$ECACCESS_URLV2/transfers"
```

returns

```
{
  "transferId" : "32595048",
  "success" : "yes"
}
code=200
```

All parameters for the ECTrans Request are listed in table 15.

To restart an ECTrans Transfer:

```
> curl $ECACCESS_OPTS -X PUT -H "Token: ahs4w294" -H "Content-Type: application/json" \
-d '{ \
  "remoteLocation":"test2", \
  "sourceFileName":"ecmwf.properties", \
  "gatewayName":"ecaccess.ecmwf.int", \
  "keepInSpool":true}' \
"$ECACCESS_URLV2/transfers/32595048"
```

returns

```
{
  "success" : "yes"
}
code=200
```

For a list of parameters for the restart of an ECTrans Request see table 15.

To delete an ECTrans Transfer:

```
> curl $ECACCESS_OPTS -X DELETE -H "Token: ahs4w294" "$ECACCESS_URLV2/transfers/32595048"
```

Type	Name	Comment
String	gatewayName	default current Gateway
String	remoteLocation	in the format association-name[@protocol]
String	sourceFileName	
String	targetFileName	
Integer	retryCount	
Integer	retryFrequency	frequency of retries in seconds (default: 600 seconds)
String	maximumDuration	e.g. 10m or 1h, ;0 to deactivate (default 12h)
String	minimumDuration	e.g. 30s or 10m, ;0 to deactivate (default 10m)
Long	minimumRate	bytes/s (default 10240)
Integer	priority	
String	lifeTime	in [w]eeks, [d]ays, [h]ours, [m]inutes or [s]econds (e.g. 1w or 2d)
Boolean	deleteSource	default FALSE
Boolean	keepInSpool	default FALSE
String	userMailAddress	
Boolean	sendMailOnSuccess	
Boolean	sendMailOnFailure	
Boolean	sendMailOnRetry	
String	ifTargetFileExists	'reject', 'append', 'resume' or 'overwrite'

Table 15: Parameters for the ECtrans Request. For additional information please see the [ecaccess-ectrans-request manpage](#).

returns

```
{
  "success" : "yes"
}
code=200
```

7 The Web server

The ECaccess gateway HTTP/S interface allows Member States to manage their job submissions and file transfers from their Web browser, e.g. Firefox, Mozilla or Internet Explorer. This section gives an overview of what this interface provides and how it works. Please note that only interactive authentication as described in section 3 is supported.

The main purpose of the HTTP/S plugin is to provide easy access and monitoring for on-line users. For use from within shell scripts (batch), most of those features are also provided through the FTP plugin and are described in the previous sections.

7.1 Authentication

Assuming that the Member State ECaccess gateway (see section 2.2) runs on the server "ecaccess.meteo.ms", users connect to the application by pointing their Web browser at "http://ecaccess.meteo.ms:9080/" and will be redirected to the login page. Note that the default HTTP port number used for ECaccess is 9080.

By giving an ECMWF user identifier and a passcode, the user is authenticated and routed to a personal page; a user context is maintained for the subsequent operations from his browser. Users have the ability to request everything available from their account, until the time allocation expires or the "logout" option from the "Account" menu is selected.

Users connecting for the first time to the login page of the Web server will receive a security alert from their browser. This is normal; users have to accept the HTTP/S plugin certificate as a trusted certificate to allow the encryption of communications.

The procedure to trust the certificate depends on the browser:

- If using Internet Explorer, you will receive a security alert. You will be given an option to view the certificate. Select it, and then select the "install certificate" option. Follow the instructions to install the certificate. Once you have returned to the security alert box, select the "Accept" option.
- If using Firefox or Mozilla, you will receive a security alert. Follow the instructions in the alert box to accept the certificate as certified. In the last dialogue box you will be given an option to accept this certificate for all your sessions. Select it.

Once this procedure is complete, your future connections to the HTTP/S plugin will not produce any security alerts.

7.2 Features

After successful authentication users are redirected from the login page to the main page, from which they will be provided with a menu including available operations described in this section.

Note that the ECaccess gateway administrator can set up the HTTP/S plugin to secure only the login process. Therefore, when redirected from the secured login page to the unsecured main page you may receive a security alert. This is a normal message; just select the "Accept" option to continue.

The main page provides the following options (organized through menu entries in the left margin):

Browsing menu

- Browse files: the user can browse through his ECHOME, ECSCRATCH or ECFS files and directories.
- Delete files: users can select files to be deleted from the different places listed above.
- Copy files: users can copy files between two domains (files can be copied from an ECSCRATCH directory to an ECFS directory, for example).
- Transfer files: users can use their browser facilities to transfer files between their computer and their ECHOME, ECSCRATCH or ECFS directories; files are transferred over an FTP connection.
- Add scripts to the job list: users can select one or several scripts and add them to their job list for later submission. Users may continue browsing files, adding more scripts to their basket.
- Select scripts for submission: users can select one or several scripts for immediate submission.
- Request secure file transfers: users can select files to be sent via their transfer spool (equivalent of the TSUB command of the FTP plugin or of the `ecaccess-ectrans-request` Web Toolkit command or of the `ectrans` command on the systems at ECMWF).

Queues/Jobs menu

- Browse queues: users can browse through the "ecgate" queues to select a target queue for their next job request.
- Browse basket: users can select scripts from their basket for their next job request.
- Submit new jobs: users can specify complementary parameters related to the execution and confirmed action of their request. The application then submits the job request, which is sent to the job spool (equivalent of the JREQ command of the FTP plugin).

Monitoring menu

- Monitor job submissions: see section [8.1](#).
- Monitor secure file transfers: see section [8.2](#).
- Browse the events history: the history allows saving details (date, name and summary) concerning each event for later consultation by users themselves.

Account menu

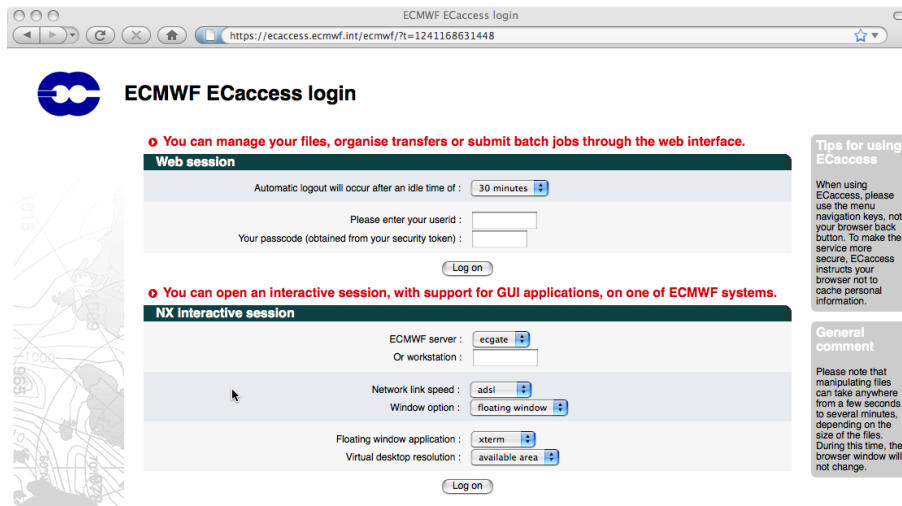
- Access the ECtrans configuration: the user can define the mapping between his ECMWF user identifier and his local user identifiers. He can also check his available protocols.
- Request a new ECaccess Certificate: the user can download a new ECaccess Certificate (description and purpose of these Certificates are discussed in section [3](#)).
- Logout: the user context is deleted and the browser is sent back to the login page.

7.3 Users views

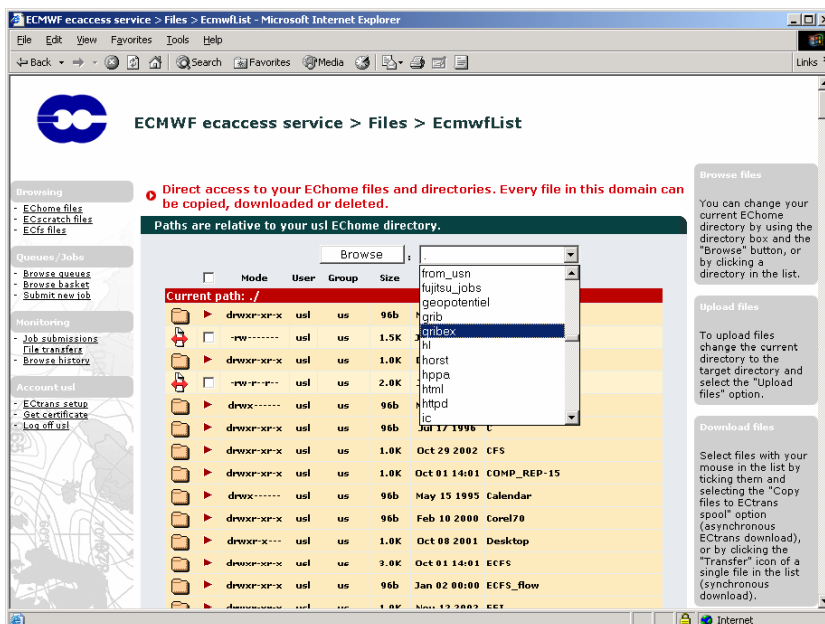
The following snapshots illustrate a typical interactive session a user could have using the web interface.

Different browsers on different operating systems may have different presentations of the same page.

First, under the heading “Web session”, login by providing your ECMWF user identifier and your passcode. You may modify the default value of 30 minutes to a greater value, if you plan to use the service with breaks of more than 30 minutes.

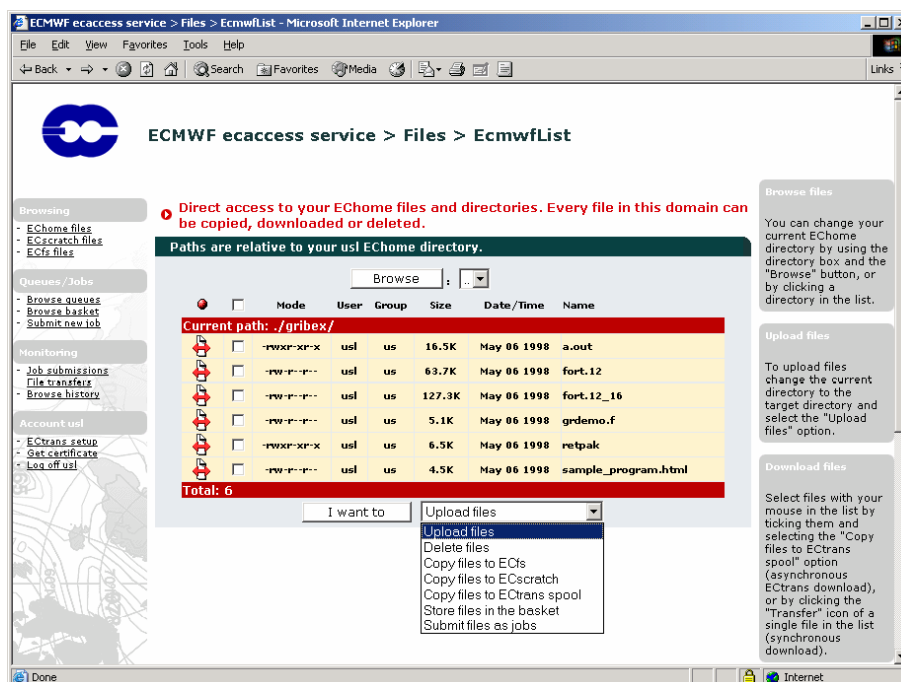


Once authenticated, your browser is redirected to the main page containing the menu described in the previous section (the default option is “Browsing > ECHOME files”). To browse other directories from your home directory, select a target directory and press the “Browse” button.

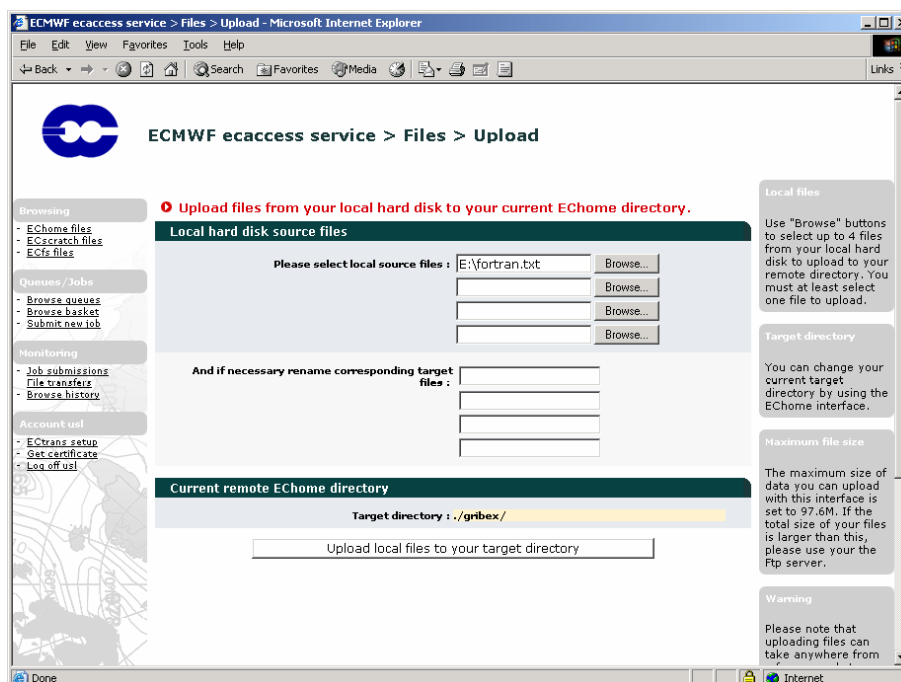


To download a file from your current directory (./gribex in this case), click the transfer icon of the target file

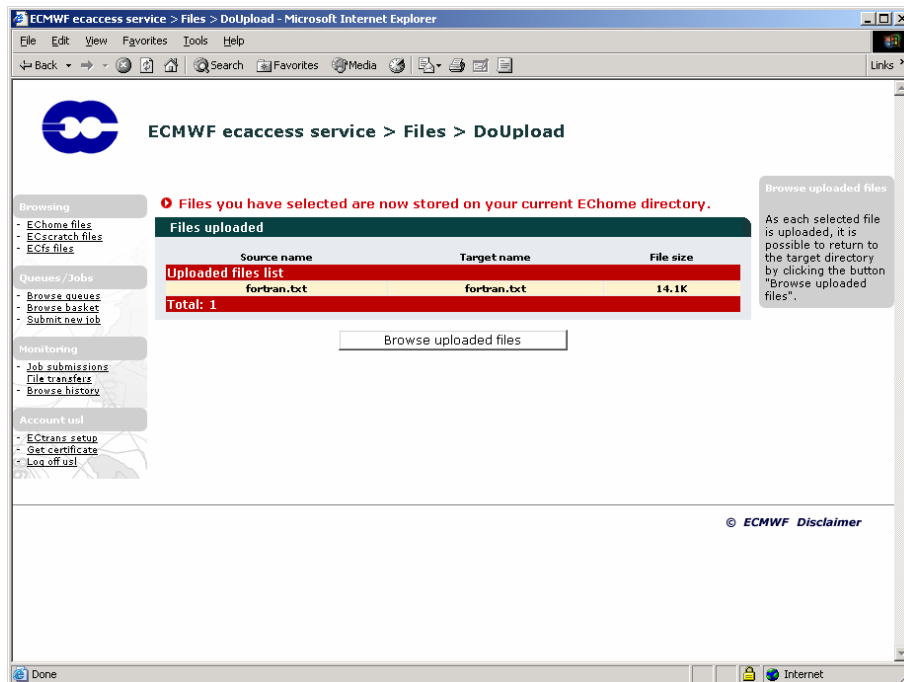
in the list. To upload a file into your current directory select the "Upload files" option and click the "I want to" button.



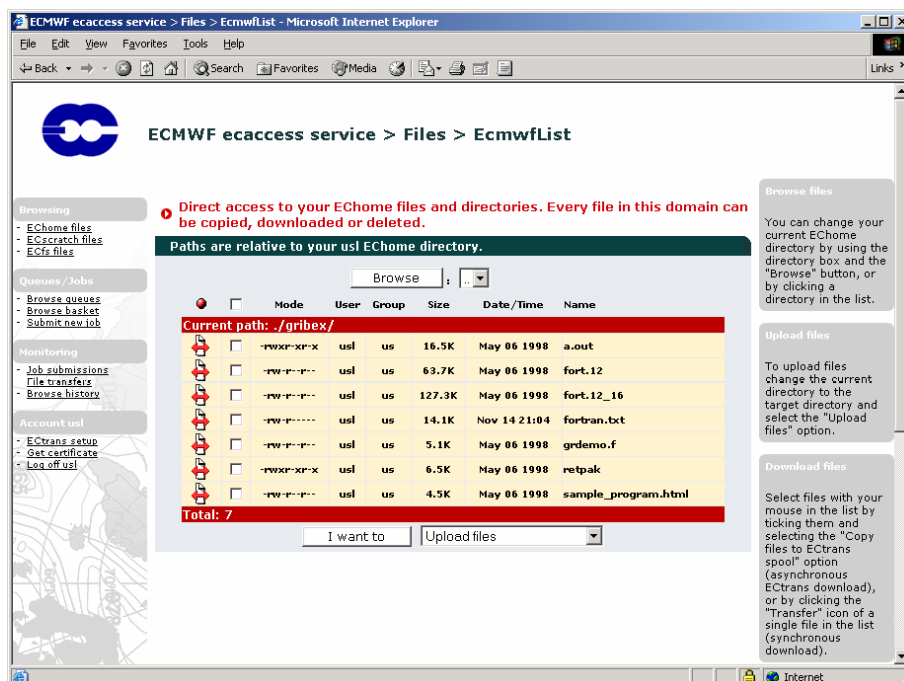
Click the "Browse" button and select the file (E:\fortran.txt) you want to upload to your current directory (you may repeat the operation three times if you want to transfer more than one file). Then click the "Upload local files to your target directory".



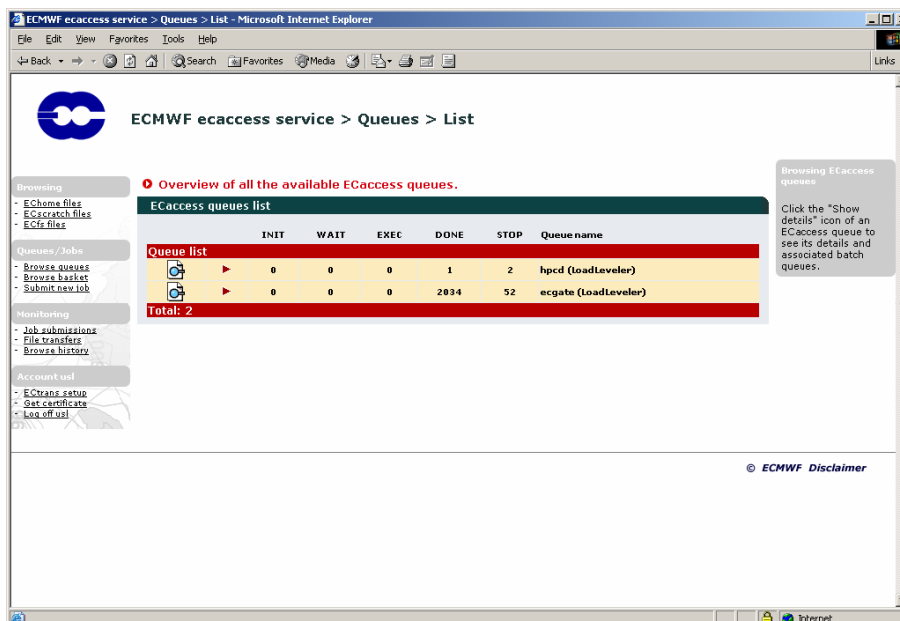
Once uploaded, a summary is printed to inform you of the size of the files uploaded. You may click the "Browse uploaded files" to return to your current directory (where your files have been uploaded).



You can see the “fortran.txt” file is now stored in your current directory. You can continue browsing directories and repeat the operation as many times as you need. To submit a job, you should first choose which system at ECMWF you want to use. To have a list of the systems at ECMWF supporting a batch service, click the “Browse queues” button.





The queues shown are known as ECAccess queues. For each of these ECAccess queues, you can click on the “show details” icon to see its associated batch queues on the system at ECMWF, e.g. below for the ECAccess queue hpcd:



ECMWF eaccess service > Queues > List

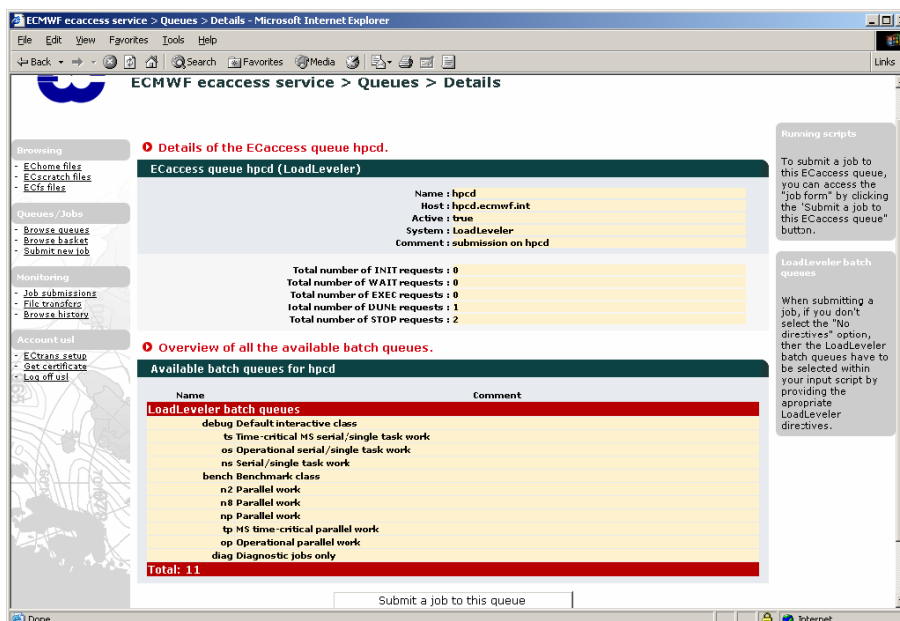
Overview of all the available EAccess queues.

EAccess queues list

	INIT	WAIT	EXEC	DONE	STOP	Queue name
 hpcd (LoadLeveler)	0	0	0	1	2	hpcd (LoadLeveler)
 ecgate (LoadLeveler)	0	0	0	2034	52	ecgate (LoadLeveler)
Total: 2						

ECMWF Disclaimer

To submit a new job, select the “Submit new job” option in the “Queues/Jobs” menu.



ECMWF eaccess service > Queues > Details

Details of the EAccess queue hpcd.

EAccess queue hpcd (LoadLeveler)

Name : hpcd
Host : hpcd.ecmwf.int
Active : true
System : LoadLeveler
Comment : submission on hpcd

Total number of INIT requests : 0
Total number of WAIT requests : 0
Total number of EXEC requests : 0
Total number of DONE requests : 1
Total number of STOP requests : 2

Overview of all the available batch queues.

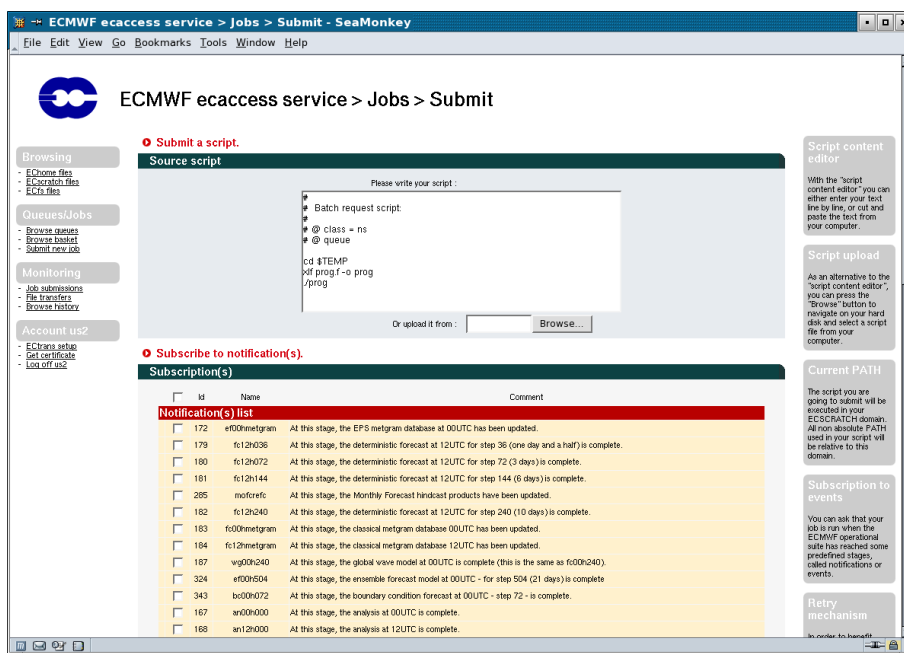
Available batch queues for hpcd

Name	Comment
LoadLeveler batch queues	
debug Default interactive class	ts Time-critical MS serial/single task work os Operational serial/single task work ns Serial/single task work
bench Benchmark class	n2 Parallel work n8 Parallel work np Parallel work
diag Diagnostic jobs only	tp MS time-critical parallel work op Operational parallel work
Total: 11	

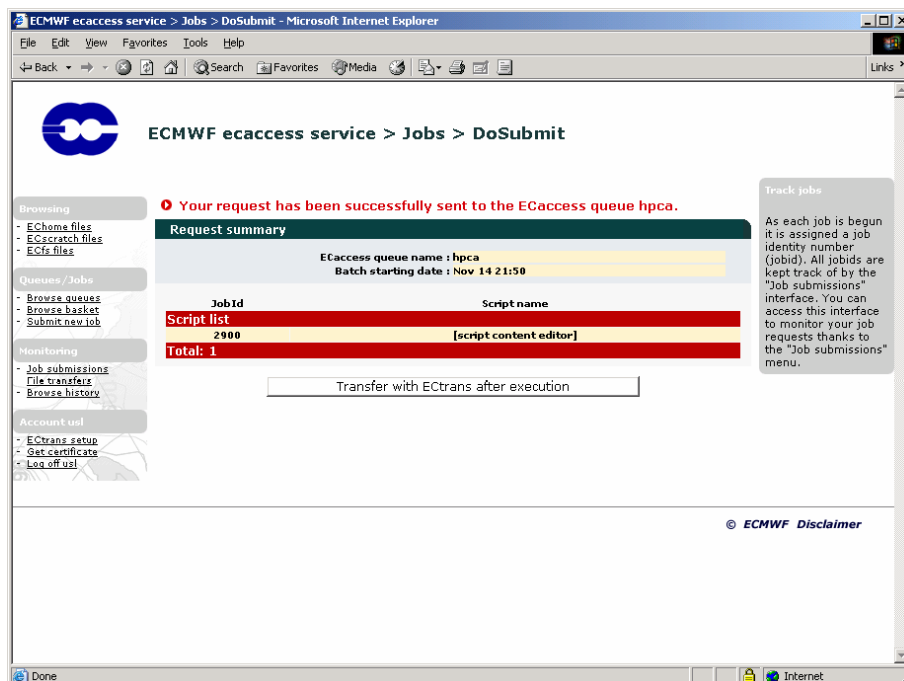
Submit a job to this queue

You may enter your script in the text area provided or select a script from your computer. Select the target queue (“hpcd” in this case). Note that the batch queue (or class) and other batch directives have to be included in your script. Alternatively, you can inform EAccess that your script does not contain batch directives. In this case, default values will be used and EAccess will fully manage your submission. Once your script is read, click the “Submit job” button to send your request to the server.

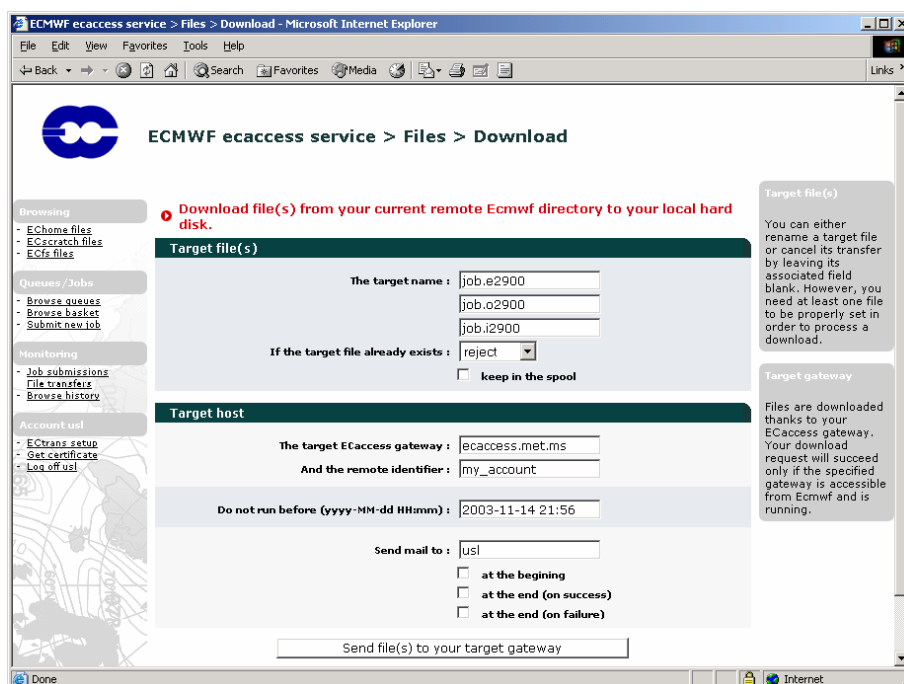
The list of notifications allows you to attach your job to one event in the ECMWF operational suite. Please refer to www.ecmwf.int/services/computing/docs/ms_items/tc_for_MS.html for further details.



Once the job is submitted, a summary screen gives you the job identifier number of your new job request. It can be used to reference the submitted job using the monitoring interface (described in the next section). If you want to arrange a secure file transfer of the result, click the “Transfer with Ectrans after execution” button.



If required, modify the default values (gateway name, user identifier) and specify the erase option of the secure file transfer (erase option is discussed in section 4.2). Then click the “Send file(s) to your target host” to proceed.



ECMWF eaccess service > Files > Download

Download file(s) from your current remote Ecmwf directory to your local hard disk.

Target file(s)

The target name :

If the target file already exists : keep in the spool

Target host

The target EAccess gateway :
 And the remote identifier :

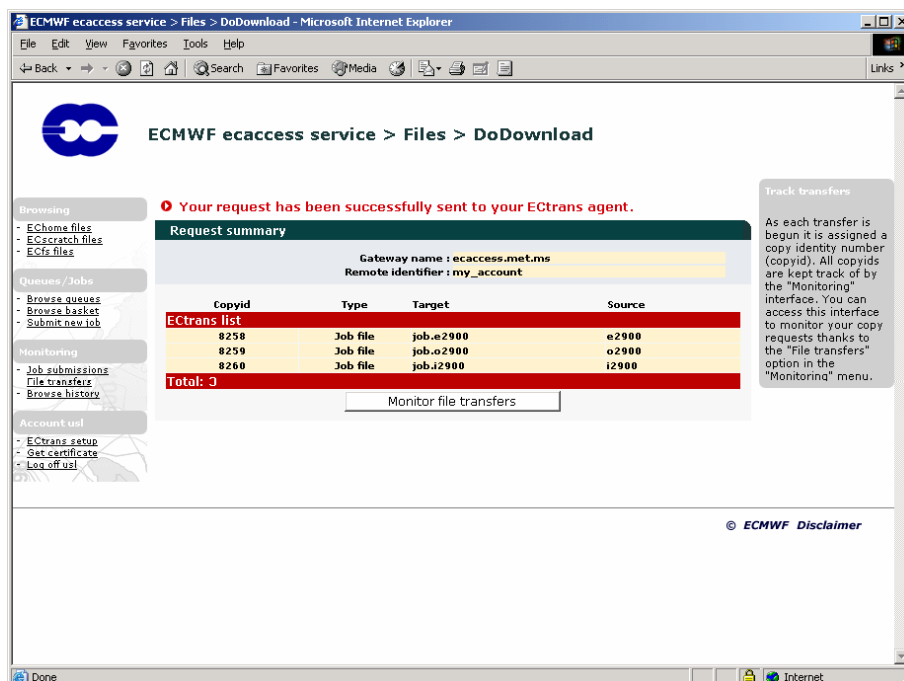
Do not run before (yyyy-MM-dd HH:mm) :

Send mail to : at the beginning
 at the end (on success)
 at the end (on failure)

Target file(s)
You can either rename a target file or cancel its transfer by leaving its associated field blank. However, you need at least one file to be properly set in order to process a download.

Target gateway
Files are downloaded thanks to your EAccess gateway. Your download request will succeed only if the specified gateway is accessible from Ecmwf and is running.

Once it is spooled, a summary screen gives you the copy identifier number of your new transfer request. It can be used to reference the secure file transfer using the monitoring interface (described in the next section).



ECMWF eaccess service > Files > DoDownload

Your request has been successfully sent to your Ectrans agent.

Request summary

Gateway name :
 Remote identifier :

Copyid	Type	Target	Source
Ectrans list			
8258	Job file	job.e2900	e2900
8259	Job file	job.o2900	o2900
8260	Job file	job.i2900	i2900
Total: 3			

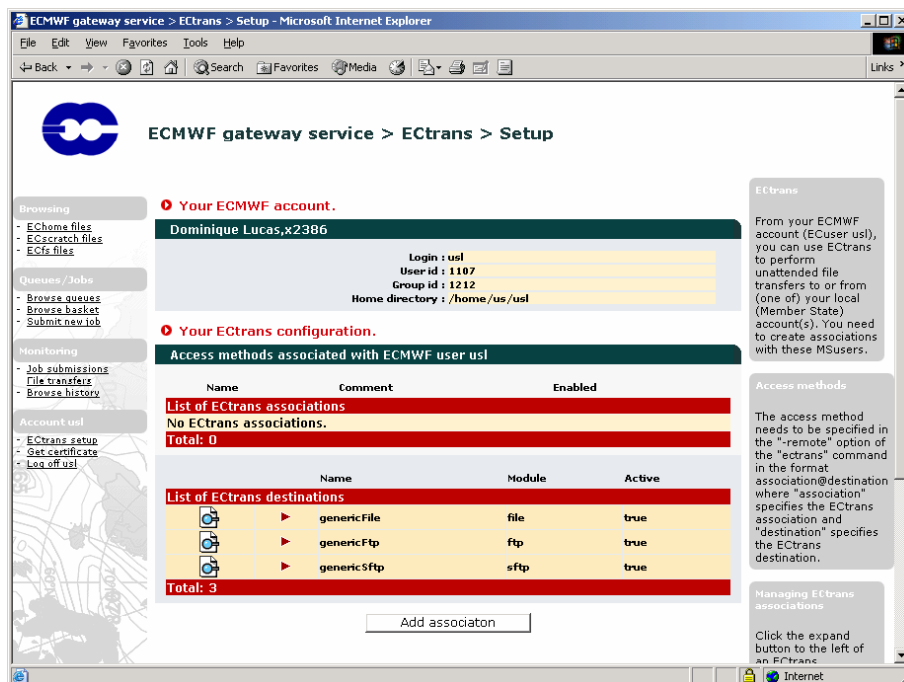
Track transfers
As each transfer is begun it is assigned a copy identity number (copyid). All copyids are kept track of by the "Monitoring" interface. You can access this interface to monitor your copy requests thanks to the "File transfers" option in the "Monitoring" menu.

© ECMWF Disclaimer

7.4 Ectrans setup

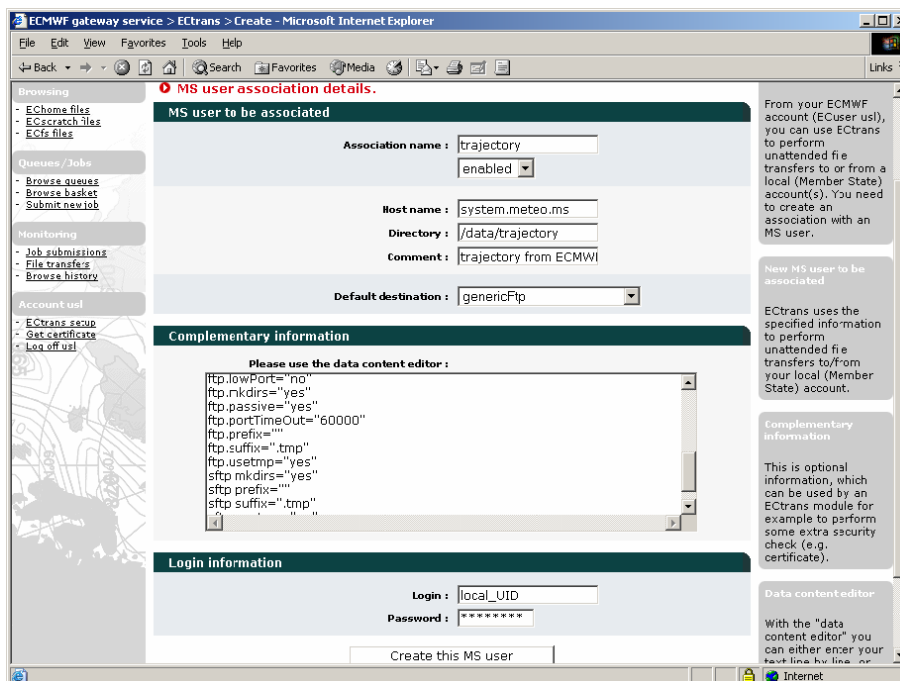
Before being able to launch unattended transfers from ECMWF (section 4) back to your site, using the command ectrans, you will have to configure an ectrans association between your ECMWF User ID and the

remote system and user-id. This is done through the web interface, by clicking “Ectrans setup” from the lower left panel.



To create a new association, click the “Add association” button. Choose an Association name, “trajectory” in the example below. This is the name that will be used as association (previously know as 'msuser') with the ectrans command. Fill in the remaining info, giving the required information on your local system.

In the example below, we create an association named trajectory that will be used to transfer files using ftp by default to a local system named “system.meteo.ms” as a user local_UID. The data transferred will be written into the directory /data/trajectory. The local files will have a temporary suffix “.tmp” added to their names during the transfer. Note that you can change the configuration of the ectrans association by modifying the options given in the window titled “Complementary information”:



MS user association details.

MS user to be associated

Association name :

enabled

Host name :

Directory :

Comment :

Default destination :

Complementary information

Please use the data content editor :

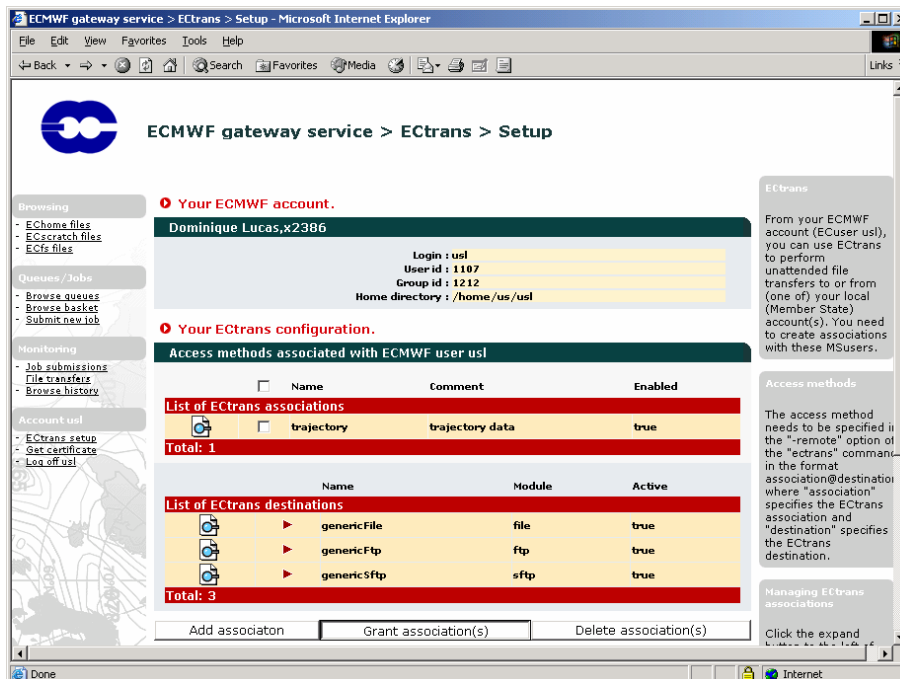
```
ftp.lowPort="no"
ftp.mkdirs="yes"
ftp.passive="yes"
ftp.portTimeOut="60000"
ftp.prefix=""
ftp.suffix=".tmp"
ftp.setmp="yes"
sftp.mkdirs="yes"
sftp.prefix=""
sftp.suffix=".tmp"
```

Login information

Login :

Password :

When you have entered all the information for your association, click the button “Create this MS user”. A new association has been defined for you. Please note that (between all users) an association name can be defined only once per gateway. You can define more associations, e.g. to transfer files from ECMWF to different systems or other local UIDs. You can also allow other users at ECMWF to transfer files with ectrans to your association. To do this, click the “Grant Association(s)” button:



ECMWF gateway service > Ectrans > Setup

Your ECMWF account.

Dominique Lucas, x2386

Login : usl

User id : 1107

Group id : 1212

Home directory : /home/us/usl

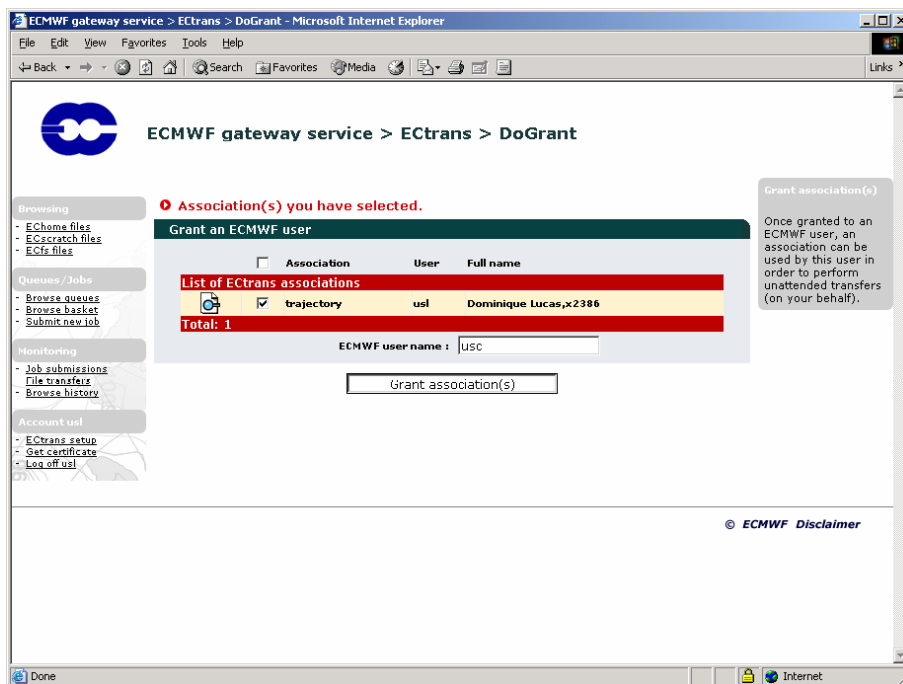
Your Ectrans configuration.

Access methods associated with ECMWF user usl

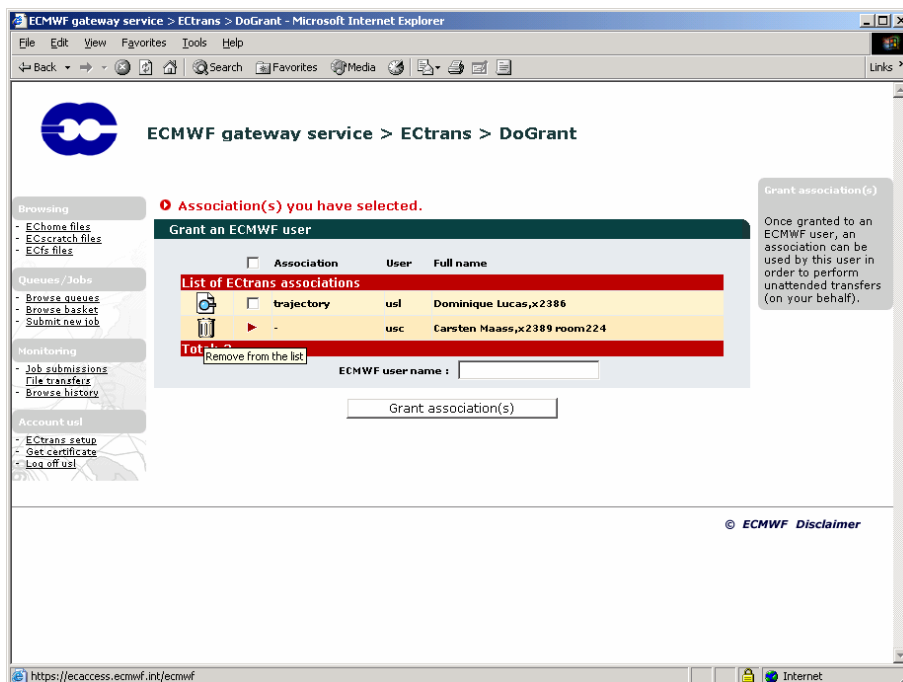
<input type="checkbox"/>	Name	Comment	Enabled
<input type="checkbox"/>	trajectory	trajectory data	true
Total: 1			

<input type="checkbox"/>	Name	Module	Active
<input type="checkbox"/>	genericFile	file	true
<input type="checkbox"/>	genericFtp	ftp	true
<input type="checkbox"/>	genericSftp	sftp	true
Total: 3			

Select the association to which you want to give access to another user. Enter the ECMWF user name. Then grant the association.



The UID and name of the person you have given access to the association is now added to the list. To remove an entry from the list, click the “Remove from the list” icon on the left:



7.5 NX service

A service using the NX technology allows users to run at ECMWF X Window based applications like Metview, XCdp, or a simple xterm.

The easiest way to use this service is via a web browser, see section 7.5.1.

It is also possible to connect using a standalone NX client application completely independent of any web browser, see section 7.5.2. A similar service is available through the ECaccess gateway “msaccess.ecmwf.int” and through your local gateway provided that you have installed the ECaccess gateway v3.3.0 at least.

NX allows you to run remote X Window sessions even across slow or low-bandwidth network connections, making it possible to start sessions from clients running on Windows, Linux, Mac OS X and Solaris platforms.

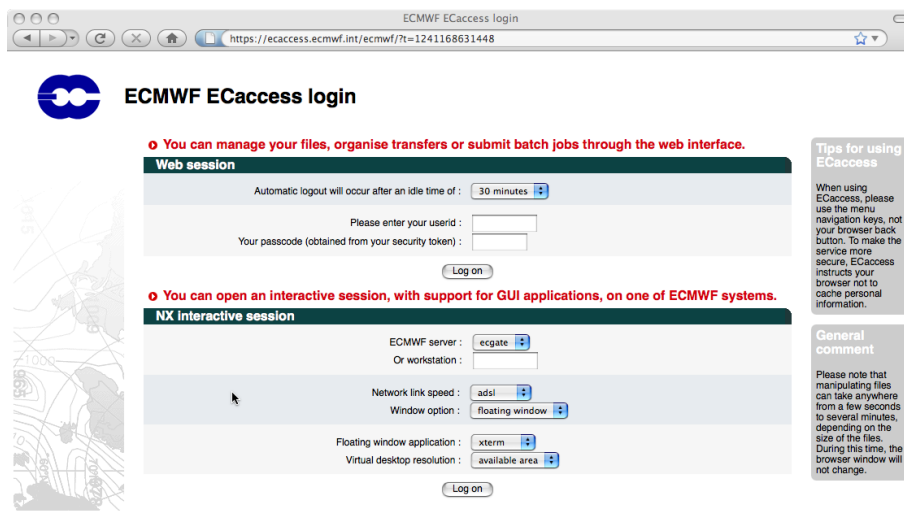
Thanks to exclusive X protocol compression techniques and an integrated set of proxy agents, NX improves the power of the X Window System to transparently run graphical desktops and applications through the network. Even on slow or low-bandwidth network connections, you can get a fast response thanks to the NX lazy encoding algorithm and NX capability to automatically tune itself to network bandwidth and latency parameters.

In addition NX allows having both standalone X terminal and “virtual desktops” independent of the web browser session used to start them. The windows can be minimised and the web browser can even be terminated.

For more information on NX, please see www.nomachine.com/documents.php.

7.5.1 How to connect using a web browser

The easiest way to connect to ECMWF using the NX service is simply to go to: <http://ecaccess.ecmwf.int/>. You will get to a page like:



ECMWF ECaccess login

Web session

Automatic logout will occur after an idle time of: 30 minutes

Please enter your userid:

Your passcode (obtained from your security token):

Log on

NX interactive session

ECMWF server:

Or workstation:

Network link speed:

Window option:

Floating window application:

Virtual desktop resolution:

Log on

Tips for using ECaccess

When using ECaccess, please use the menu navigation keys, not your browser back button. To make the service more secure, ECaccess instructs your browser not to cache personal information.

General comment

Please note that manipulating files can take anywhere from a few seconds to several minutes, depending on the size of the files. During this time, the browser window will not change.

Using various drop down menus in the *bottom part of the page* you will be able to select the type of NX session you want to establish. Please note that your web browser needs to be Java enabled.

You can connect to both ecgate and the supercomputer using the drop down menu “ECMWF server”.

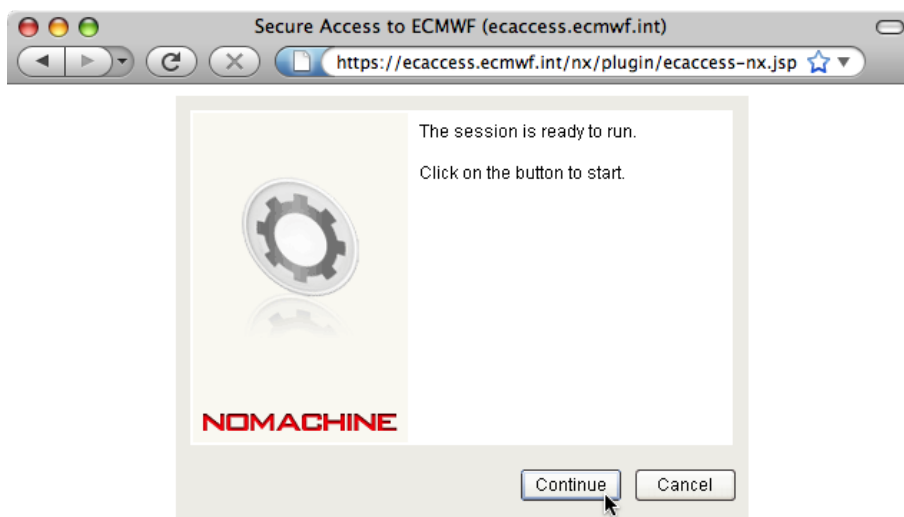
You can select the type of network link you are using with the menu “Network link speed”. This will select a number of options which should be optimal for your configuration.

You can select the type of window you want to have using the “Window option” menu: if you select “floating window” you will get a single X Window application like xterm or Metview (you can choose the application using the next menu). If, instead, you select “virtual desktop” you will get a fully working desktop using the WindowMaker window manager. In this case you can select the “Virtual desktop resolution” to be either “available area” or “full screen”.

7.5.2 Example of session starting a standalone xterm on the supercomputer

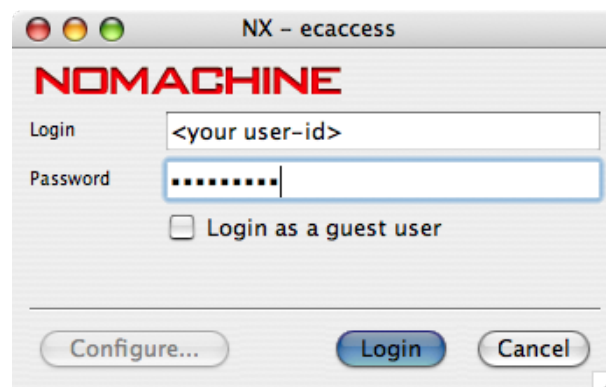
In this case you need to select “c2a” as “ECMWF server”, specify your type of network link (you can leave this to the default “ads!”), then select “floating window” as “Window option”, leave the default “Floating window application” to “xterm” and press “Log on”.

This, after some windows warning about certificates and ssh key which you need to accept, will display the following page:

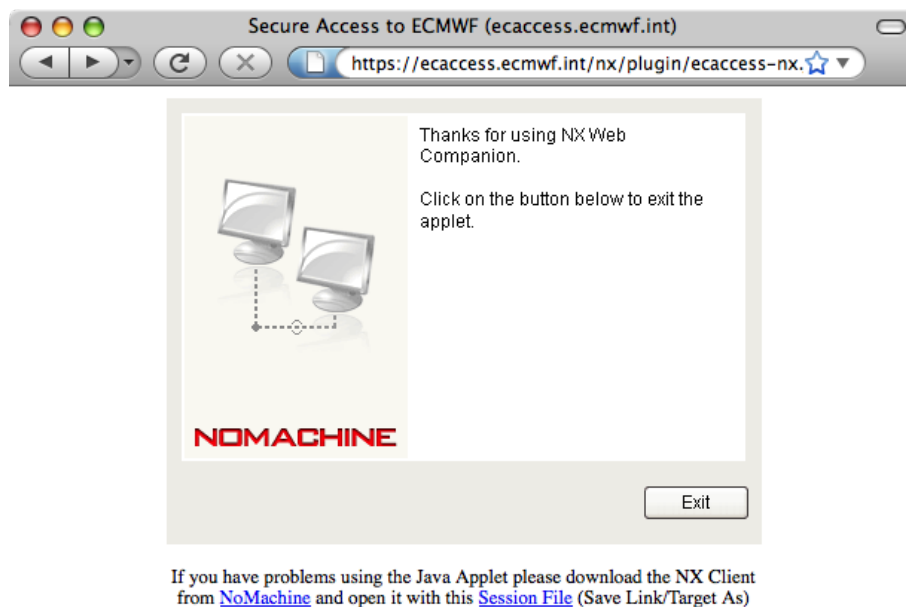


If you have problems using the Java Applet please download the NX Client from [NoMachine](#) and open it with this [Session File](#) (Save Link/Target As)

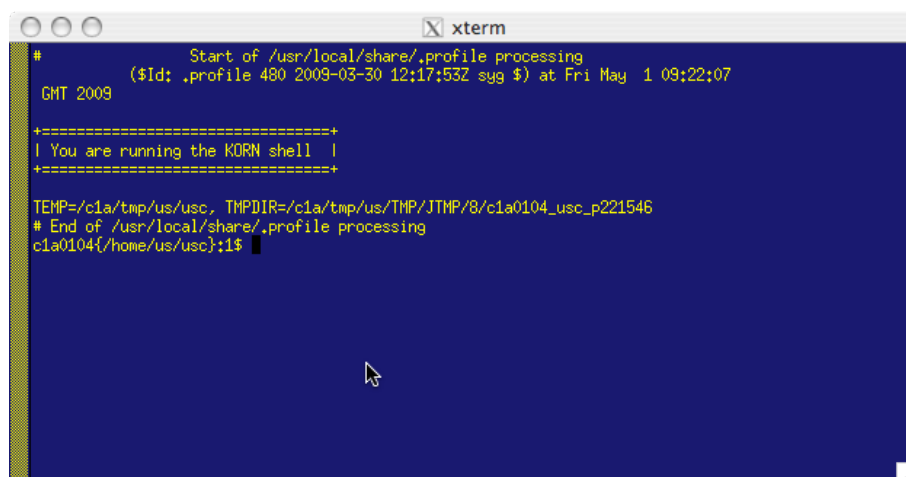
You will need to click on the “Continue” button to start the NX connection. The following window will appear:



This window allows you to enter your userid and corresponding *passcode generated by your security token*. After entering the appropriate information click on “Login” to proceed. The Java applet in the web browser will display various messages detailing the progress of the connection to ECMWF (depending on your firewall setup you may get various warning messages: you will need to authorise all sessions from anything related to NX - nxclient, nxauth, nxssh, etc) until this will be displayed in your browser:



The application you have requested to start, in this case an “xterm”, should also start as a separate X based window. You can now minimise (or even close) your web browser and start using your xterm.



7.5.3 Example of session starting a virtual desktop on ecgate

In this case select the following (for the link speed you can leave the default “adsl”):

ECMWF server :	ecgate
Or workstation :	
Network link speed :	adsl
Window option :	virtual desktop
Floating window application :	xterm
Virtual desktop resolution :	available area

and press “Log on”. The login process will be the same as the one described in the previous example but at the end the following window will appear:

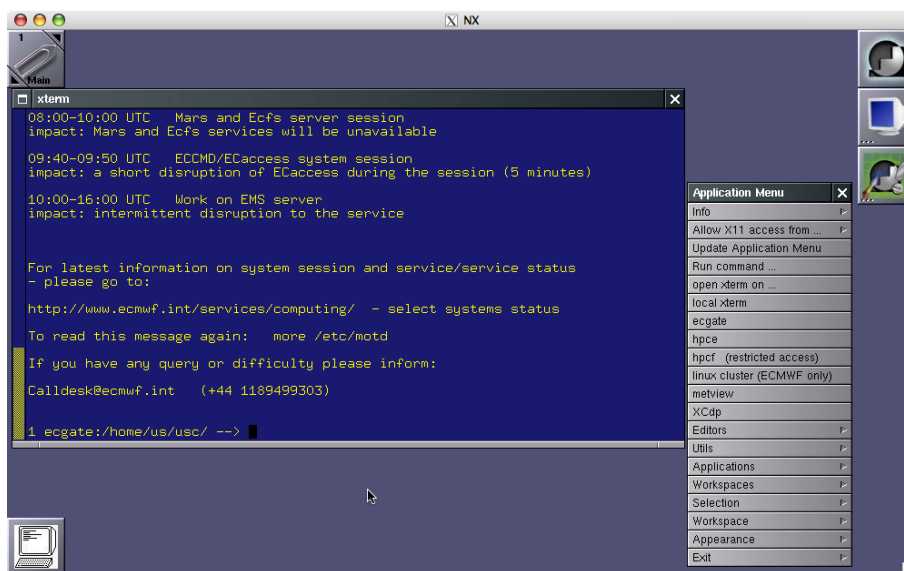


Figure 2: Virtual desktop on ecgate started using NX.

The window manager available on this desktop is called WindowMaker. By right clicking on the mouse you will get an Application Menu which allows you to start an xterm or other X based applications. The main desktop window is a standalone X Window and can be minimised. If you prefer, you can start a virtual desktop in full screen mode by choosing the “Virtual desktop resolution” option “full screen”. Section 7.5.5 below describes the usage of WindowMaker in more detail.

7.5.4 How to connect using a standalone NX client

In addition to using the web browser based access to ECMWF via NX described previously, you can also download a standalone NX client. To do this, go to www.nomachine.com/download.php and select the NX client for your platform. The installation is quite straightforward and is described in more detail at www.nomachine.com/documents/client/install.php. You can then use the “Download session file” option available through the web interface:

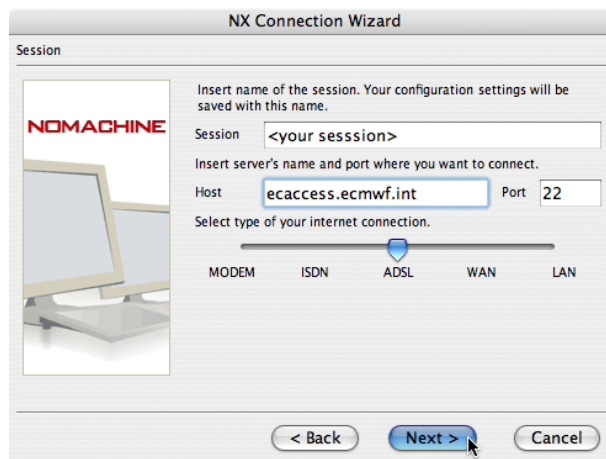


This URL allows you to download a complete configuration file which can be used with your standalone NX client. You can have multiple configuration files, say one for a standalone xterm on ecgate and another one for a full virtual desktop still on ecgate, and then select the appropriate one from your NX client.

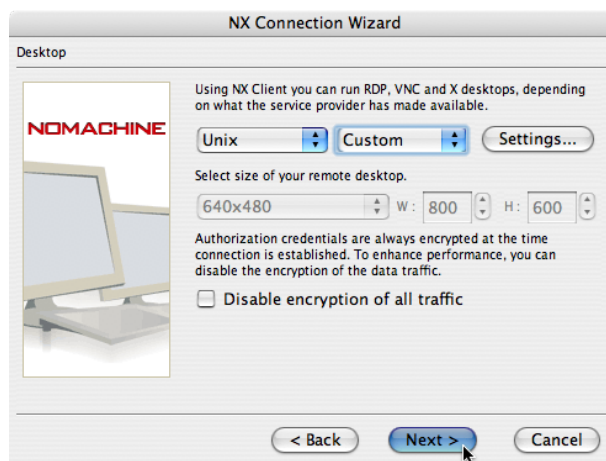
Alternatively, you can use the NX client “Wizard” to setup your own configuration as described in the NX client documentation available at www.nomachine.com/documents/configuration/client-guide.php We recommend using this option for advanced users only. We also recommend that you first look at one of the configuration files which you can obtain by downloading the “session file”. The first time you start the NX client the following window will appear:



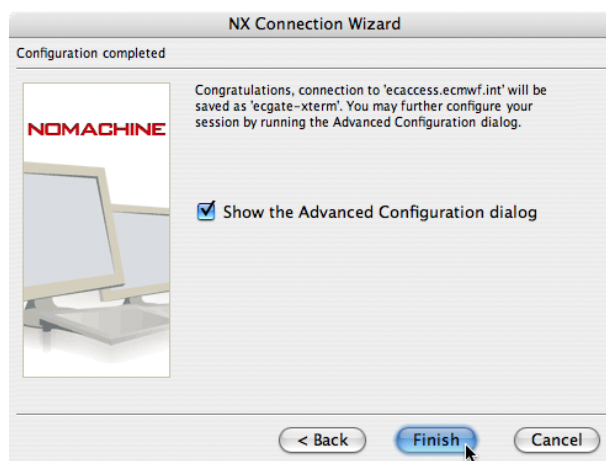
You will have to click “Next” where you will be asked to enter the name of your NX session (in the example `<your session>`) and the host to connect to. You will have to enter the ECaccess host name “ecaccess.ecmwf.int” as host:



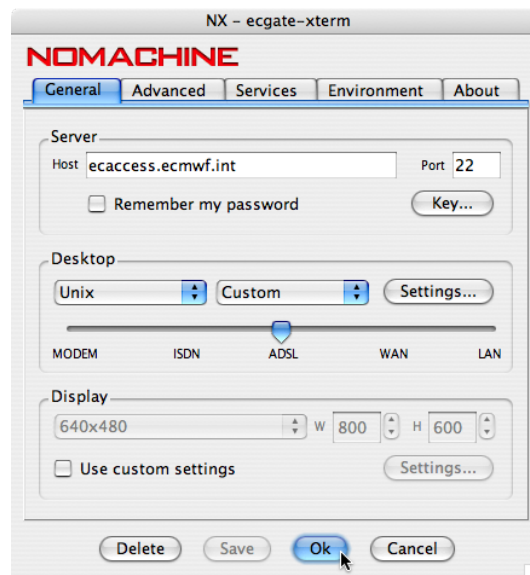
You will then get the following window where you can choose you type of desktop. You will need to choose “Unix” and “Custom”:



Click on “Next” to get the following window:



Check the “Show the Advanced Configuration dialog” box and click the Finish button. You will get the following window:



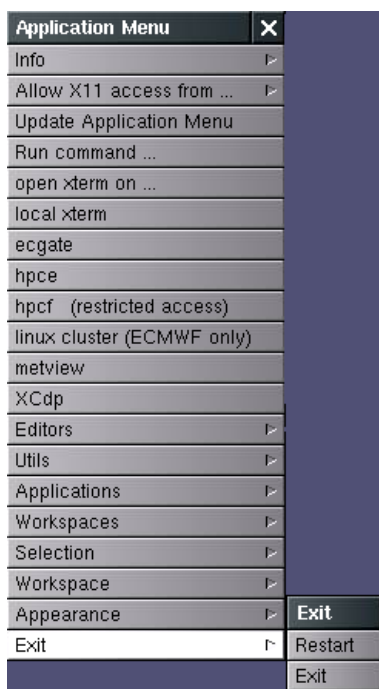
If you then click “Ok” you will be able to start your session. In this case you will get a standalone xterm on ecgate. Depending on your firewall setup you may get various warning messages. You will need to authorise all sessions from anything related to NX (nxclient, nxauth, nxssh, etc).

7.5.5 WindowMaker overview

WindowMaker is a popular window manager for the X Window System, allowing graphical applications to be run on Unix-like operating-systems. It is designed to emulate NeXT’s GUI as an OpenStep-compatible environment and has been described as “one of the most useful and universal window managers available.” WindowMaker has a reputation for being fast, efficient and highly stable and is very popular among open source solutions for use on both newer and older machines. More information on WindowMaker can be found at http://en.wikipedia.org/wiki/Window_Maker and www.windowmaker.info.

WindowMaker is the window manager which is used when you connect with NX to either ecgate or the supercomputer and select the “virtual desktop” option. For example, when you connect to ecgate using the virtual desktop you will get a desktop as shown in figure 2.

The main customisation which has been implemented is a specific “Application Menu” which you can obtain when right-click (opposite mouse button for left-handed mouse) on the desktop. The menus on ecgate and the supercomputer are designed to be very similar with the one on ecgate offering more choices regarding the available applications. The usage of the menus should be quite straightforward. To terminate a WindowMaker session you need to select the “Exit” option from the menu:



8 Monitoring tools

The purpose of the monitoring interface is to provide Member States users with information concerning:

- Job requests referenced by the job identifier number, which is returned by the `ecaccess-job-submit` command (see section 5).
- Secure file transfer requests referenced by the copy identifier number, which is returned by the `ecaccess-ectrans-` command (see section 5) or the `ectrans` command (see section 4.2).

The monitoring interface is accessible through the ECaccess HTTP/S plugin, which supports the interactive method of authentication described in section 3.

Procedures to login and use this plugin are discussed in the previous section. The following discussion assumes that you are connected.

8.1 Monitoring batch job submissions

To access this interface, select the option “Job submissions” in the “Monitoring” menu.

ECMWF ecaccess service > Jobs > Track

Use this interface to track jobs you have submitted to ECMWF.

Jobs submitted by usl

<input type="checkbox"/>	JobId	ECaccess queue	Date/Time	Status
<input type="checkbox"/>	3146	ecgate1 (NQS)	Nov 15 21:33	WAIT
<input type="checkbox"/>	1163	hpca (LoadLeveler)	Nov 13 10:14	STOP
<input type="checkbox"/>	1162	hpca (LoadLeveler)	Nov 13 09:29	STOP
<input type="checkbox"/>	1161	hpca (LoadLeveler)	Nov 13 00:27	STND
Total: 4				

Delete selected

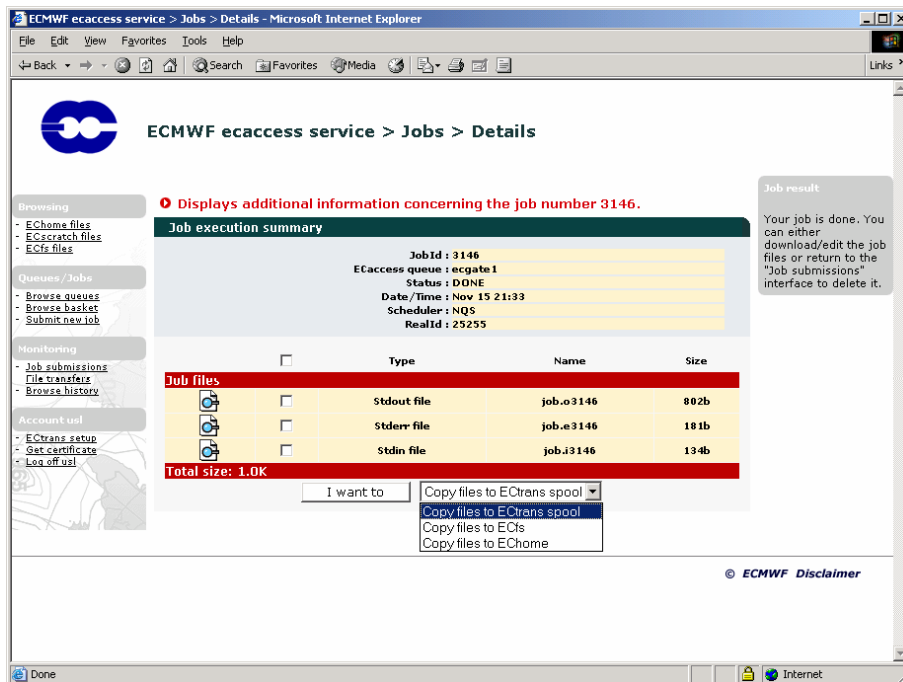
Jobs list
All jobs are kept track of by their job identity number. The job status can be either "INIT", "WAIT", "EXEC", "DONE" or "STOP".

Job result
To get more details about a job, select it with your mouse in the list by clicking the "Track job" icon.

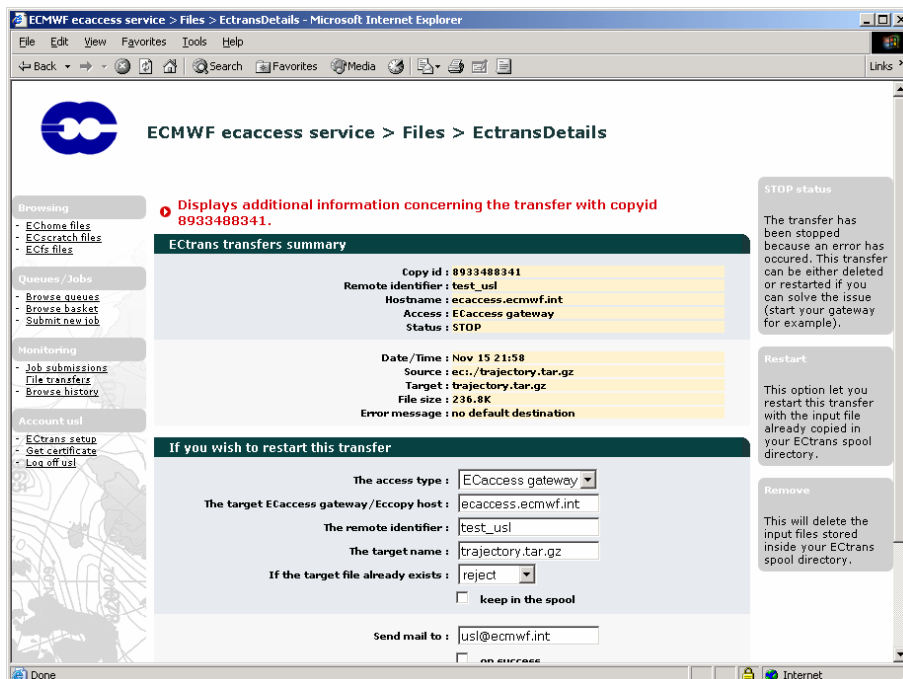
Delete
You can delete one or more than one job by ticking the job(s) in the list and clicking the "Delete selected" button.

© ECMWF Disclaimer

Your submitted jobs are listed. You are informed of the status of the jobs (meanings of the different values are provided in the help tips). You can use the “show details” icon to get more information about a job. For example, if a job submission failed, you can get the reason for this failure by looking at the job details. Once a job is marked as “DONE” you can select it with your mouse to see its output.



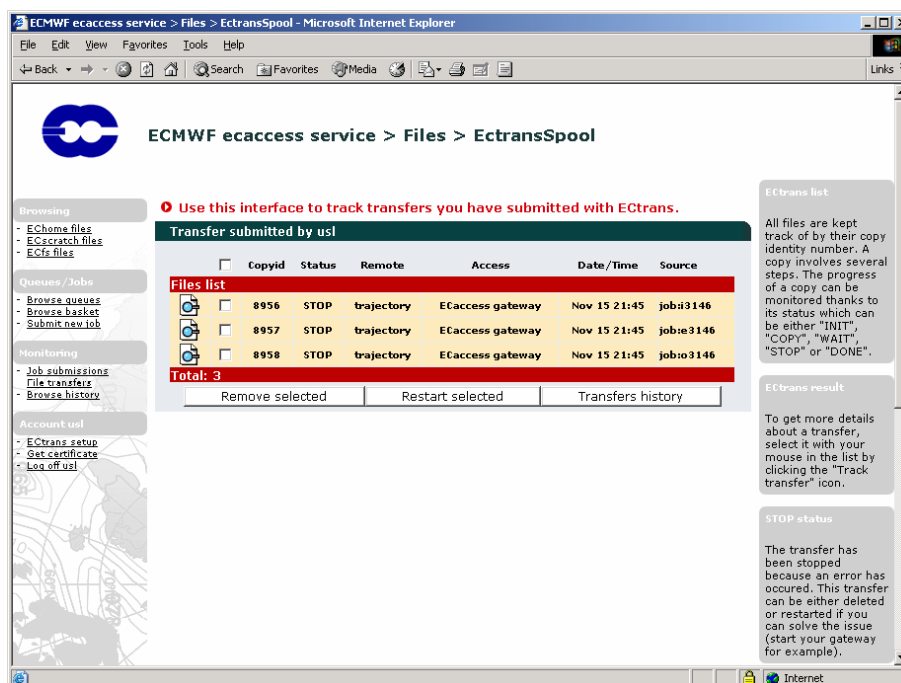
You can view the content of the output, error or input files associated with the job. You can also choose not to consult these files on-line but copy them to one of your directories or get them using the secure file transfer feature. Use the "I want to" button for this purpose. To edit one of them, just click the edit icon on the corresponding line.



You may use the cut and paste function of the operating system to get the complete file, or just read it on-line.

8.2 Monitoring ectrans file transfers

To access this interface, select the option “File transfers” in the “Monitoring” menu.

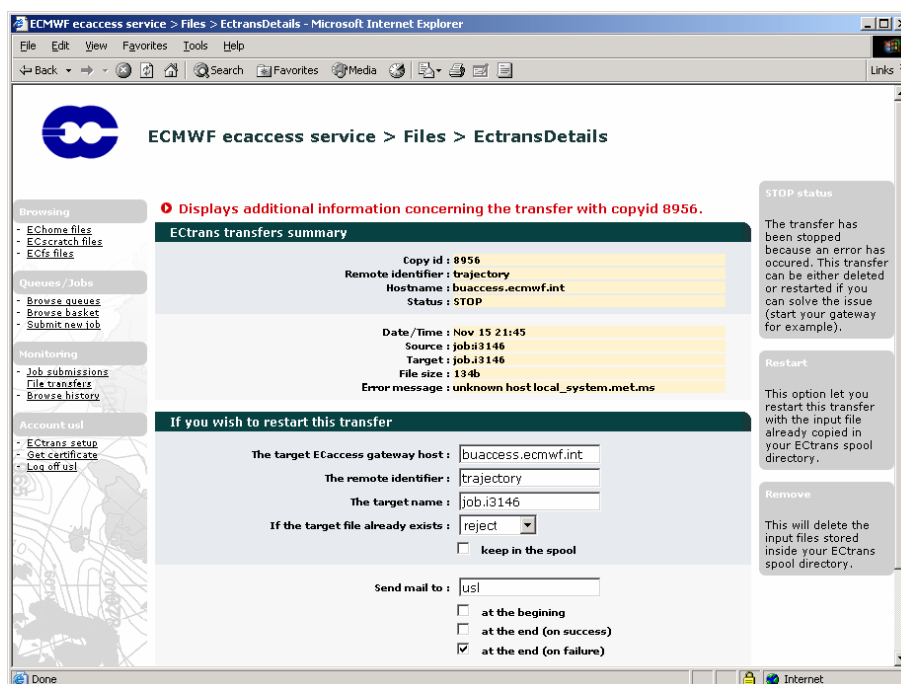


The screenshot shows the 'ECMWF ectrans service > Files > EctransSpool' interface. A red banner at the top reads: 'Use this interface to track transfers you have submitted with Ectrans.' Below this is a table titled 'Transfer submitted by us!' with columns: Copyid, Status, Remote, Access, Date/Time, and Source. The table contains three rows of data, all with a status of 'STOP'.

Copyid	Status	Remote	Access	Date/Time	Source
8956	STOP	trajectory	Ectrans gateway	Nov 15 21:45	job:3146
8957	STOP	trajectory	Ectrans gateway	Nov 15 21:45	job:3146
8958	STOP	trajectory	Ectrans gateway	Nov 15 21:45	job:3146

Below the table, it says 'Total: 3' and provides buttons for 'Remove selected', 'Restart selected', and 'Transfers history'. On the right side, there are help boxes: 'Ectrans list' explaining that files are tracked by copy identity number and progress can be monitored; 'Ectrans result' explaining how to get more details; and 'STOP status' explaining that a transfer is stopped due to an error and can be restarted.

A list displays your transfer requests. You are informed of the status of the transfers (meanings of the different values are provided in the help tips). Once a transfer is marked as “DONE” or “STOP” you can select it with your mouse and obtain the following screen:



The screenshot shows the 'ECMWF ectrans service > Files > EctransDetails' interface. A red banner at the top reads: 'Displays additional information concerning the transfer with copyid 8956.' The main content area is titled 'Ectrans transfers summary' and displays the following details:

- Copy id : 8956
- Remote identifier : trajectory
- Hostname : buaccess.ecmwf.int
- Status : STOP
- Date/Time : Nov 15 21:45
- Source : job:3146
- Target : job:3146
- File size : 134b
- Error message : unknown host local_system.met.ms

Below this summary is a section titled 'If you wish to restart this transfer' with several input fields and checkboxes:

- The target Ectrans gateway host :
- The remote identifier :
- The target name :
- If the target file already exists :
 - keep in the spool
- Send mail to :
 - at the beginning
 - at the end (on success)
 - at the end (on failure)

On the right side, there are help boxes: 'STOP status' explaining the error and restart options; 'Restart' explaining that the input file is already copied; and 'Remove' explaining that it will delete files from the spool directory.

9 The Telnet server

The Telnet plugin (**available only on MSgateways**) allows Member State users to log into their shell account at ECMWF and execute commands directly on an ECMWF machine. When contacting the EAccess service with telnet, you will see something like:

```
Connected to ecaccess.
Escape character is '^]'.
Authorized access only.

*****
For further information, read the EAccess
documentation at:
-> http://www.ecmwf.int/services/ecaccess/

You can also use EAccess to load/download
files from your EHome, ECscratch or ECfs
directories using the EAccess FTP server:
-> ftp://uid@ecaccess.ecmwf.int/

Use your UID and the SecurID code to login!
*****

TelnetPlugin v3.0.0_2005010701
login: uid
Passcode:*****
```

The prompt is for your ECMWF user identifier. You will then be prompted for your passcode (obtained by entering your PIN number into your security token), and then you will get a UNIX prompt, typically '\$' or '%'. A login with telnet puts you automatically in your home directory.

Note that a different message may be displayed during your login procedure, as this message is customisable by the gateway administrator. This option gives the opportunity to broadcast important notes to Member State users (availability of a new product, disruptions planned for maintenance purposes, etc.).

The Telnet plugin supports only the interactive method of authentication described in section 3.

Note that the gateway at ECMWF will close telnet sessions idle for 6 hours. If you use a Member State EAccess gateway, note that the default port number used by ecaccess is 9023. You'll therefore have to run:

```
-> telnet ecaccess.meteo.ms 9023
```

If you are not sure which port number to use please check with your local Eaccess administrator.

10 The SSH server

The SSH plugin (part of the gateway) allows Member State users to log into their shell account at ECMWF and execute commands directly on “ecgate”. The first time you use SSH to ECaccess, you will see something like:

```
-> ssh uid@ecaccess.ecmwf.int
The authenticity of host 'ecaccess.ecmwf.int (193.61.196.110)' can't be
established.
DSA key fingerprint is 9e:e3:f0:12:f5:08:61:d8:55:89:1a:40:e6:18:b8:42.
Are you sure you want to continue connecting (yes/no)? yes
*****
For further information, read the ECaccess
documentation at:
-> http://www.ecmwf.int/services/ecaccess/

You can also use ECaccess to load/download
files from your EChome, ECscratch or ECfs
directories using the ECaccess FTP server:
-> ftp://uid@ecaccess.ecmwf.int/

Use your UID and the SecurID code to login!
*****

Password authentication
uid's password *****
```

You will then be prompted for your passcode (obtained by entering your PIN number into your security token), and then will get a UNIX prompt, typically '\$' or '%'. A login with SSH puts you automatically in your home directory on ecgate.

Note that a different message may be displayed during your login procedure, as this message is customisable by the gateway administrator. This option gives the opportunity to broadcast important notes to Member State users (availability of a new product, disruptions planned for maintenance purposes, etc.).

The SSH plugin supports only the interactive method of authentication described in section 3.

Note that the gateway at ECMWF will close SSH sessions idle for 6 hours.

Note also that if you use a Member State ECaccess gateway, there is no need to use ssh, as the connection between the MS gateway and ECMWF is already secure. Using telnet will do. If you decide to use your MS gateway (and your gateway administrator has opened this service), you may need to contact port number 9022, like in:

```
-> ssh -p 9022 -l uid ecaccess.meteo.ms
```

11 X11 connections

The X11 plugin (part of the gateway) allows Member State users who have an X11 server running on their workstation to log into their shell account and start X11 applications directly on ECMWF systems.

First, users must check that their DISPLAY environment variable is properly set up on their workstation:

```
-> echo $DISPLAY
hostname:0.0
```

The content of this variable is the name of the display to which X11 applications will connect (usually the name of the user workstation).

If users have a server access control program for X, they must add the gateway hostname to their host list allowed to make connections to their X11 server, e.g., assuming that the Member State EAccess gateway (see section 2.2) runs on the server “ecaccess.meteo.ms”, with the “xhost” command

```
-> xhost +ecaccess.meteo.ms
ecaccess.meteo.ms being added to access control list
```

The MS gateway is then authorized to open connections to their X11 server. Note that the “xhost” command is only required for telnet, not for ssh.

After these preliminary settings you should be able to request an X11 proxy via your telnet or SSH connection. Each subsequent X11 application started from this xterm window (including new xterm) will make connections to your X11 server.

11.1 Starting xterm within a SSH session

By connecting to “ecaccess.ecmwf.int” with the SSH plugin described in section 10, after having been validated with your security token, you will first have to select the system at ECMWF to access.

Note that you may have to use “ssh -X” to open the X11 tunnel.