@@@@@@@@@

./rev\_1.01.sql

@@@@@@@@@@

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

job\_id CHAR(36) NOT NULL PRIMARY KEY

,job\_state VARCHAR(9) NOT NULL

,job\_params VARCHAR(250)

,user\_dn TEXT NOT NULL

,user\_cred VARCHAR(250)

,reason VARCHAR(255)

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job.

--

file\_id INTEGER NOT NULL UNIQUE AUTO\_INCREMENT

-- job\_id (used in joins with file table

,job\_id CHAR(36)

NOT NULL

REFERENCES t\_job(job\_id)

-- the state of this file

,file\_state VARCHAR(29)

NOT NULL

-- The Source

,source\_surl TEXT

NOT NULL

-- The Destination

,dest\_surl TEXT

NOT NULL

-- The transport specific paramaters for the file transfer

,file\_params VARCHAR(250)

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

,reason VARCHAR(255)

-- why it failed

,duration INTEGER,

PRIMARY KEY (job\_id, file\_id),

KEY(file\_id)

) TYPE=INNODB;

DROP TABLE IF EXISTS t\_logical;

CREATE TABLE t\_logical (

--

file\_id INTEGER NULL PRIMARY KEY

REFERENCES t\_file(file\_id)

-- job\_id

,job\_id CHAR(36)

NOT NULL

REFERENCES t\_job(job\_id)

-- The generalised Source

,source TEXT NOT NULL

-- The Destination Surl or Site

,destination TEXT NOT NULL

) TYPE=INNODB;

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

name VARCHAR(255)

,domain\_a VARCHAR(100)

,domain\_b VARCHAR(100)

,contact VARCHAR(255)

,bandwidth INTEGER

,state VARCHAR(30)

) TYPE=INNODB;

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

--

-- A logging table which tracks interactions with the t\_job and t\_file table

--

DROP TABLE IF EXISTS t\_job\_log;

CREATE TABLE t\_job\_log (

job\_id CHAR(36)

NOT NULL

,file\_id INTEGER

,job\_state VARCHAR(9)

,file\_state VARCHAR(9)

,timestamp DATE

,dn TEXT

) TYPE=INNODB;

-- Schema version

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

@@@@@@@@@

./rev\_1.02.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

,domain\_a VARCHAR(100) NOT NULL

,domain\_b VARCHAR(100) NOT NULL

,contact VARCHAR(255)

-- bandwidth in Mb/s

,bandwidth INTEGER

,state VARCHAR(30)

,last\_active DATE

) TYPE=INNODB;

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(9) NOT NULL

--

-- transport specific parameters

,job\_params VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32) REFERENCES t\_channel(channel\_name)

,reason VARCHAR(255)

,submit\_time DATE

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

--

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job.

file\_id INTEGER NOT NULL UNIQUE AUTO\_INCREMENT

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- the state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The transport specific paramaters for the file transfer

,file\_params VARCHAR(250)

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

,reason VARCHAR(255)

-- the time at which the file was transferred

,transfer\_time DATE

-- the transfer duration in seconds

,duration INTEGER,

PRIMARY KEY (job\_id, file\_id),

KEY(file\_id)

) TYPE=INNODB;

--

-- t\_logical stores the

--

DROP TABLE IF EXISTS t\_logical;

CREATE TABLE t\_logical (

--

-- the file\_id index for this logical file

file\_id INTEGER NULL PRIMARY KEY

REFERENCES t\_file(file\_id)

--

-- job\_id index for that file.

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The generalised logical Source

,source TEXT NOT NULL

--

-- The Destination Surl or logical destination site

,destination TEXT NOT NULL

) TYPE=INNODB;

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

--

-- A logging table which tracks interactions with the t\_job and t\_file table

--

DROP TABLE IF EXISTS t\_job\_log;

CREATE TABLE t\_job\_log (

job\_id CHAR(36) NOT NULL

,file\_id INTEGER

,job\_state VARCHAR(9)

,file\_state VARCHAR(9)

,timestamp DATE

,dn TEXT

) TYPE=INNODB;

-- Schema version

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

@@@@@@@@@

./rev\_1.03.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Close domain name: "cern.ch"

,domain\_a VARCHAR(100) NOT NULL

--

-- Far domain name: "rl.ac.uk"

,domain\_b VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsbile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(9) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL UNIQUE

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written INTEGER

--

-- the nominal size of the file (bytes)

,filesize INTEGER

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

,PRIMARY KEY (job\_id, file\_id),

KEY(file\_id)

) TYPE=INNODB;

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (1,2,2);

@@@@@@@@@

./rev\_1.04.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Close domain name: "cern.ch"

,domain\_a VARCHAR(100) NOT NULL

--

-- Far domain name: "rl.ac.uk"

,domain\_b VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsbile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(9) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL UNIQUE

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- File params - deprecated; only for backwards compatibility. Will be removed in next major update.

,file\_params VARCHAR(255)

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written INTEGER

--

-- the nominal size of the file (bytes)

,filesize INTEGER

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

,PRIMARY KEY (job\_id, file\_id),

KEY(file\_id)

) TYPE=INNODB;

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (1,3,0);

@@@@@@@@@

./rev\_1.05.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Close domain name: "cern.ch"

,domain\_a VARCHAR(100) NOT NULL

--

-- Far domain name: "rl.ac.uk"

,domain\_b VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsbile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(9) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL UNIQUE

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- File params - deprecated; only for backwards compatibility. Will be removed in next major update.

,file\_params VARCHAR(255)

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- Number of retries this file has had (0 for first attempt)

,num\_retries INTEGER

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

,PRIMARY KEY (job\_id, file\_id),

KEY(file\_id)

) TYPE=INNODB;

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (1,3,0);

@@@@@@@@@

./rev\_1.06.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Close domain name: "cern.ch"

,domain\_a VARCHAR(100) NOT NULL

--

-- Far domain name: "rl.ac.uk"

,domain\_b VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsbile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL UNIQUE

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- File params - deprecated; only for backwards compatibility. Will be removed in next major update.

,file\_params VARCHAR(255)

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- Number of retries this file has had (0 for first attempt)

,num\_retries INTEGER

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

,PRIMARY KEY (job\_id, file\_id),

KEY(file\_id)

) TYPE=INNODB;

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (1,3,0);

@@@@@@@@@

./rev\_1.07.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Close domain name: "cern.ch"

,domain\_a VARCHAR(100) NOT NULL

--

-- Far domain name: "rl.ac.uk"

,domain\_b VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsbile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source SE name - the source cluster name

,source\_se VARCHAR(250)

--

-- Dest SE name - the destination cluster name

,dest\_se VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL UNIQUE

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- File params - deprecated; only for backwards compatibility. Will be removed in next major update.

,file\_params VARCHAR(255)

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- Number of retries this file has had (0 for first attempt)

,num\_retries INTEGER

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

,PRIMARY KEY (job\_id, file\_id),

KEY(file\_id)

) TYPE=INNODB;

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (1,3,0);

@@@@@@@@@

./rev\_1.08.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

);

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id VARCHAR(255) NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time TIMESTAMP(0)

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written INTEGER

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.09.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

);

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time TIMESTAMP(0)

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written INTEGER

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.10.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

);

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- Number of Failures

,num\_failures INTEGER

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time TIMESTAMP(0)

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written INTEGER

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.11.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active TIMESTAMP(0)

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

);

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time TIMESTAMP(0)

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- Number of Failures

,num\_failures INTEGER

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time TIMESTAMP(0)

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp TIMESTAMP

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.12.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

);

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- Number of Failures

,num\_failures INTEGER

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.13.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

);

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- Number of Failures

,num\_failures INTEGER

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.14.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

);

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

--

-- Internal job parameters,used to pass job specific data from the

-- WS to the agent

,internal\_job\_params VARCHAR(255)

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- Number of Failures

,num\_failures INTEGER

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.15.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

);

--

-- Store VO ACL

--

DROP TABLE IF EXISTS t\_vo\_acl;

CREATE TABLE t\_vo\_acl (

--

-- the name of the VO

vo\_name VARCHAR2(32) NOT NULL

--

-- The principal name

,principal VARCHAR2(255)

--

-- Set Primary Key

, PRIMARY KEY (vo\_name, principal)

);

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

--

-- Internal job parameters,used to pass job specific data from the

-- WS to the agent

,internal\_job\_params VARCHAR(255)

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- Number of Failures

,num\_failures INTEGER

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_vo\_acl indexes:

-- t\_vo\_acl(vo\_name,principal) is primary key

CREATE INDEX vo\_acl\_vo\_name ON t\_vo\_acl(vo\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.16.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

) TYPE=INNODB;

--

-- Store VO ACL

--

DROP TABLE IF EXISTS t\_vo\_acl;

CREATE TABLE t\_vo\_acl (

--

-- the name of the VO

vo\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255)

--

-- Set Primary Key

, PRIMARY KEY (vo\_name, principal)

) TYPE=INNODB;

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

--

-- Internal job parameters,used to pass job specific data from the

-- WS to the agent

,internal\_job\_params VARCHAR(255)

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- Number of Failures

,num\_failures INTEGER

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_vo\_acl indexes:

-- t\_vo\_acl(vo\_name,principal) is primary key

CREATE INDEX vo\_acl\_vo\_name ON t\_vo\_acl(vo\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,0,0);

@@@@@@@@@

./rev\_1.17.sql

@@@@@@@@@@

--

-- channels in the system

--

DROP TABLE IF EXISTS t\_channel;

CREATE TABLE t\_channel (

--

-- Name of the channel

channel\_name VARCHAR(32) PRIMARY KEY NOT NULL

--

-- Source Site name

,source\_site VARCHAR(100) NOT NULL

--

-- Destination Site name

,dest\_site VARCHAR(100) NOT NULL

--

-- Email contact of the channel responsibile

,contact VARCHAR(255)

--

-- Maximum bandwidth, capacity, in Mbits/s

,bandwidth INTEGER

--

-- The target number of concurrent streams on the network

,nostreams INTEGER

--

-- The target number of concurrent files on the network

,nofiles INTEGER

--

-- The target throughput for the system (Mbits/s)

,nominal\_throughput INTEGER

--

-- The state of the channel ("Active", "Inactive", "Drain", "Stopped")

,channel\_state VARCHAR(30)

--

-- The date the channel was last active

,last\_active DATETIME

) TYPE=INNODB;

--

-- The Channel VO share table stores the percentage of the channel resources

-- available for a VO

--

DROP TABLE IF EXISTS t\_channel\_vo\_share;

CREATE TABLE t\_channel\_vo\_share (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The name of the VO

,vo\_name VARCHAR(50) NOT NULL

--

-- The percentage of the channel resources associated to the VO

,channel\_share INTEGER DEFAULT 0

--

-- Set primary key

,PRIMARY KEY (channel\_name, vo\_name)

) TYPE=INNODB;

--

-- Store Channel ACL

--

DROP TABLE IF EXISTS t\_channel\_acl;

CREATE TABLE t\_channel\_acl (

--

-- the name of the channel

channel\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255) NOT NULL

--

-- Set Primary Key

,PRIMARY KEY (channel\_name, principal)

) TYPE=INNODB;

--

-- Store VO ACL

--

DROP TABLE IF EXISTS t\_vo\_acl;

CREATE TABLE t\_vo\_acl (

--

-- the name of the VO

vo\_name VARCHAR(32) NOT NULL

--

-- The principal name

,principal VARCHAR(255)

--

-- Set Primary Key

, PRIMARY KEY (vo\_name, principal)

) TYPE=INNODB;

--

-- t\_job contains the list of jobs currently in the transfer database.

--

DROP TABLE IF EXISTS t\_job;

CREATE TABLE t\_job (

--

-- the job\_id is a IETF UUID in string form.

job\_id CHAR(36) NOT NULL PRIMARY KEY

--

-- The state a job is currently in

,job\_state VARCHAR(13) NOT NULL

--

-- Transport specific parameters

,job\_params VARCHAR(250)

--

-- Source Site/SE name - the source cluster name

,source VARCHAR(250)

--

-- Dest Site/SE name - the destination cluster name

,dest VARCHAR(250)

--

-- the DN of the user starting the job - they are the only one

-- who can sumbit/cancel

,user\_dn TEXT NOT NULL

--

-- the DN of the agent currently servicing the job

,agent\_dn TEXT NOT NULL

--

-- the user credentials passphrase. This is passed to the movement service in

-- order to retrieve the appropriate user proxy to do the transfers

,user\_cred VARCHAR(250)

--

-- Blob to store user capabilites and groups

,voms\_cred BLOB

--

-- The VO that owns this job

,vo\_name VARCHAR(50)

--

-- the name of the transfer channel for this job

,channel\_name VARCHAR(32)

REFERENCES t\_channel(channel\_name)

--

-- The reason the job is in the current state

,reason VARCHAR(255)

--

-- The date that the job was submitted

,submit\_time DATETIME

--

-- Priority for Intra-VO Scheduling

,priority INTEGER DEFAULT 3

--

-- Internal job parameters,used to pass job specific data from the

-- WS to the agent

,internal\_job\_params VARCHAR(255)

) TYPE=INNODB;

--

-- t\_file stores the actual file transfers - one row per source/dest pair

--

DROP TABLE IF EXISTS t\_file;

CREATE TABLE t\_file (

-- file\_id is a unique identifier for a (source, destination) pair with a

-- job. It is created automatically.

--

file\_id INTEGER NOT NULL AUTO\_INCREMENT PRIMARY KEY

--

-- job\_id (used in joins with file table)

,job\_id CHAR(36) NOT NULL

REFERENCES t\_job(job\_id)

--

-- The state of this file

,file\_state VARCHAR(29) NOT NULL

--

-- The Source Logical Name

,logical\_name TEXT

--

-- The Source

,source\_surl TEXT NOT NULL

--

-- The Destination

,dest\_surl TEXT NOT NULL

--

-- The agent who is transferring the file. This is only valid when the file

-- is in 'Active' state

,agent\_dn TEXT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the file is in this state

,reason VARCHAR(255)

--

-- Number of Failures

,num\_failures INTEGER

--

-- the nominal size of the file (bytes)

,filesize BIGINT

--

-- the user-defined checksum of the file "checksum\_type:checksum"

,checksum VARCHAR(100)

--

) TYPE=INNODB;

--

-- t\_transfer table stores the data related to a file transfer

--

DROP TABLE IF EXISTS t\_transfer;

CREATE TABLE t\_transfer (

--

-- transfer request id

request\_id VARCHAR(255) NOT NULL

--

-- Identifier of the file to transfer

,file\_id INTEGER NOT NULL

REFERENCES t\_file(file\_id)

--

-- transfer identifier within the request. It could be used for ordering

,transfer\_id INTEGER NOT NULL

--

-- The state of this file

,transfer\_state VARCHAR(29) NOT NULL

--

-- The Source TURL

,source\_turl TEXT

--

-- The Destination TURL

,dest\_turl TEXT

--

-- the time at which the file was transferred

,transfer\_time DATETIME

--

-- the transfer duration in seconds

,duration FLOAT

--

-- the number of bytes written to the destination

,bytes\_written BIGINT

--

-- The class for the reason field

,reason\_class VARCHAR(32)

--

-- The reason the transfer is in this state

,reason TEXT

--

-- Set primary key

,PRIMARY KEY (request\_id, file\_id)

);

--

-- t\_trace table traces the transfers' execution

--

DROP TABLE IF EXISTS t\_trace;

CREATE TABLE t\_trace (

--

-- message\_id is a unique identifier for the entry.

-- It is created automatically.

trace\_id INTEGER NOT NULL PRIMARY KEY

--

-- transfer request id

,request\_id VARCHAR(255)

--

-- Identifier of the file to transfer

,file\_id INTEGER

--

-- trace timestamp

,trace\_timestamp DATETIME

--

-- trace content

,content TEXT

--

-- Set Constraint

,FOREIGN KEY (request\_id, file\_id)

REFERENCES t\_transfer(request\_id,file\_id)

);

--

-- t\_agent table traces the agents' state

--

DROP TABLE IF EXISTS t\_agent;

CREATE TABLE t\_agent (

--

-- The Agent name. In case of FTS or FPS agent, the agent name should follow

-- the convention 'vo:<vo\_name>'; in case of Channel agent, the name should be

-- 'channel:<channel\_name>'

agent\_name VARCHAR(255) NOT NULL PRIMARY KEY

--

-- The agent DN

,agent\_dn TEXT

--

-- The name of the host where the agent is installed

,agent\_host VARCHAR(255)

--

-- The agent type.Can be 'channel','fts','fps'

,agent\_type VARCHAR(8)

--

-- Agent last active tiestamp

,last\_active DATETIME

--

-- Agent state. Can be 'Started', 'Stopped'

,agent\_state VARCHAR(16)

--

-- Agent version

,agent\_version VARCHAR(16)

--

-- Email contact of the agent responsbile

,contact VARCHAR(255)

);

--

--

-- Index Section

--

--

-- t\_channel indexes:

-- t\_channel(channel\_name) is primary key

CREATE INDEX channel\_channel\_state ON t\_channel(channel\_state);

-- t\_channel\_vo\_share indexes:

-- t\_channel\_vo\_share(channel\_name,vo\_name) is primary key

CREATE INDEX channel\_vo\_share\_channel\_name ON t\_channel\_vo\_share(channel\_name);

-- t\_channel\_acl indexes:

-- t\_channel\_acl(channel\_name,principal) is primary key

CREATE INDEX channel\_acl\_channel\_name ON t\_channel\_acl(channel\_name);

-- t\_vo\_acl indexes:

-- t\_vo\_acl(vo\_name,principal) is primary key

CREATE INDEX vo\_acl\_vo\_name ON t\_vo\_acl(vo\_name);

-- t\_job indexes:

-- t\_job(job\_id) is primary key

CREATE INDEX job\_job\_state ON t\_job(job\_state);

CREATE INDEX job\_channel\_name ON t\_job(channel\_name);

CREATE INDEX job\_vo\_name ON t\_job(vo\_name);

-- t\_file indexes:

-- t\_file(file\_id) is primary key

CREATE INDEX file\_job\_id ON t\_file(job\_id);

CREATE INDEX file\_file\_state ON t\_file(file\_state);

-- index up to 1000 for >MySQL 4.1.20

CREATE INDEX file\_source\_surl ON t\_file(source\_surl(255));

-- t\_transfer indexes:

-- t\_transfer(transfer\_unique\_id) is primary key

-- Probably unique index is not need it: commented out

-- CREATE UNIQUE INDEX transfer\_unique\_idx ON t\_transfer(request\_id,transfer\_id) TABLESPACE &1;

CREATE INDEX transfer\_request\_id ON t\_transfer(request\_id);

CREATE INDEX transfer\_file\_id ON t\_transfer(file\_id);

CREATE INDEX transfer\_transfer\_state ON t\_transfer(transfer\_state);

-- t\_trace indexes:

-- t\_trace(trace\_id) is primary key

CREATE INDEX trace\_transfer\_unique\_id ON t\_trace(request\_id,file\_id);

-- t\_agent indexes:

-- t\_agent(agent\_name) is primary key

CREATE INDEX agent\_agent\_state ON t\_agent(agent\_state);

--

-- Schema version

--

DROP TABLE IF EXISTS t\_schema\_vers;

CREATE TABLE t\_schema\_vers (

major INTEGER NOT NULL,

minor INTEGER NOT NULL,

patch INTEGER NOT NULL

) TYPE=INNODB;

INSERT INTO t\_schema\_vers (major,minor,patch) VALUES (2,1,0);