



Aseco v0.6.1 beta

- Read Me -

1.About

2.Requirements

3.Installation

1. Dedicated Server
2. PHP
3. Aseco

4.Public Database Plugin

1. About
2. Installation
3. Configure

5.Local Database Plugin

1. About
2. Installation
3. Configure

6.Plugins

1. Install a new plugin
2. Remove a plugin

7.Customize

1. Chat PlugIns
2. Event oriented PlugIns

8.Chat Documentation

9.Bug Reporting

10.Change Log

11.Team

1.About

Aseco is an abbreviation for „**A**utomatic **S**erver **C**ontrol“.

And that's what it does. Aseco is a brother of FAST, which was developed to save records driven on servers in a global database. Soon it got spread through the community and players began to modify it.

Aseco should combine the power which the new dedicated server provides with the functionality of FAST. But that's not possible in every case... I had to do compromises.

For example this version is not compatible with the in game server of the TM games.

You also can use Aseco to administrate your server in game, which means you can manage it by typing commands into the chat. „/admin nextmap“ will skip the current track and force the server to load the next challenge. There aren't only admin functions, players will have their fun, too!

For more see the chapter „Chat Documentation“.

2.Requirements

You will need following things to run Aseco on your server:

- Dedicated Server (03/10/06), better (03/28/06) (www.tm-forum.com)
- PHP since version 5.0 (www.php.net)
- an Aseco server account (www.tmbase.de)
- DSL 1024+

3.Installation

3.1.Dedicated Server

As said before Aseco needs a dedicated server running to connect on. This can be explained, because the TM game servers are too old and don't have the technology of new dedicated servers. As Aseco is specialized on those new functions to provide you best performance it won't run without!

You can download the latest version here: [download TM Dedicated Server](#)

This version is compatible with all latest versions of the TM games (TMO/TMSX/TMN).

A tutorial (by Blahitis) how to configure/use the Dedicated Server you'll find here: [How To Configure the Dedicated Server](#)

3.2.Php

Aseco is completely written in Php what means for you, that you can modify it as you want to. Be creative - build up your own server!
You should have installed Php 5.0 or later.

If you don't have one of the supported versions, then you can download the latest one here: [download Php](#)

To install Php just decompress the content of the received archive into any folder (i.e.: „C:\Program Files\Php“)

Alternative you of course can use your Webserver Php version. As long as it's 5.0+

3.3.Aseco

a) Download

As it seems you already got your version of Aseco. If you didn't, need some updates or want to look for some plugins then you can visit our website here: <http://www.tmbase.de/en/downloads>

b) Create Account

First thing which you should do is to create an account.

You can do so here: www.tmbase.de

You have to log in on the site using the Community Code and your TMS/TMN/TMO Login.

Just get your Community Code by logging in on the specific site:

- www.trackmanianations.com
- www.trackmaniasunrise.com
- www.trackmania.com

We need you to do this because of security reasons and to keep our database clean. We can easily delete wrong records created from always the same account then.

You will need the login and the password again when configuring Aseco.

c) Configure

Configure "config.xml"

Aseco reads its settings out of the config.xml, which lies in the root directory of where you extracted the files of the package to (i.e.: „C:\Programs\Aseco\config.xml“).

```
6   <admins>
7     <tmlogin>$000der$f00f1o/192.168.1.33:2351</tmlogin>
8   </admins>
```

If you want to be able to use the ingame administration you have to tell Aseco who the administrators are!

Replace the black text with your TM login name.

If you should connect to your server over LAN then the dedicated server will use following login for you: „*nickname/ip:port*“.

Tip: If you want to have more than one admin then you can attach additional ones by copying the line „<tmlogin></tmlogin>“ and put your login between the two tags as you already did for the first one.

```
9   <colors>
10    <emotic>$fa0</emotic>
11    <record>$0f3</record>
12    <message>$39f</message>
13    <status>$ff0</status>
14    <server>$ff0</server>
15    <highlite>$fff</highlite>
16    <welcome>$f00</welcome>
17  </colors>
```

Here Aseco reads the colours used when writing into chat.

You can define colours by modifying the black text or creating a new one.

The syntax is always the same

„<colourname>colourcode</colourname>“.

```
18  <messages>
19    <startup>{#status}*** Aseco {#highlite}v{1}{#status} running on {#hi
20    <welcome>{#welcome>Welcome {#highlite}{1}{#welcome} - This server us
21    <record_new>{#server}>> {#highlite}{1}{#record} took the {#highlite}
22    <record_current>{#server}>> {#message}Current record on {#highlite}{
23    <record_none>{#server}>> {#message}Currently no record on {#highlite}
24    <record_error>{#server}>> {#error}Could not get records from databas
25    <ranking>{#server}>> {#message}Record rankings after this round:</ra
26    <ranking_record>{#highlite}{1}{#message}. {#highlite}{2}{#message} [{#
27    <ranking_none>{#server}>> {#message}Record rankings after this round
28  </messages>
```

You can modify the messages Aseco is using to tell players who drove a new record or how to welcome joining players.

Here you can use the colours , which you defined above.

„{#colourname}“ will be replaced by the content of
„<colourname>\$fff</colourname>“.

A little example: „{#colourname} hello“ will be converted to „\$fff hello“.

„{1}“ will be replaced by a parameter. You don't have to care about what function is behind, but you should know that if you change the position of this parameter you can change the position of the specified information which will replace it.

For instance: „{#colourname} hello {1}“ will become „\$fff hello Flo“.

If „{1}“ stands for the player's name you can move it where you want to, to change the meaning of the sentence.

You can't add new messages without changing the code of Aseco, this is only for experienced Php users.

Configure “Aseco.bat”

The last step is to configure the executable.

This .bat file will run the aseco.php in the console, which is much more comfortable than just run it in the web browser!

You will have to set the directory you installed Php into here:

```
set INSTPHP=C:\Programme\Apache2\Php5
```

replace the bold text with the absolute path to your Php folder.

Choose a Database solution

Aseco originally was developed to save records of tracks from the internet and collect them in a database.

When you start Aseco now it will not be able to save records driven on your server. You will always get a message that there's no record on the current track.

That's why I have done two plugins - to learn more about plugins see chapter 6 page 8 - which will provide the basic two types of how to save your records. You can either enable one of them or both, so they run parallel.

So you see Aseco needs a database to store records or player statistics, read on to find out which database is the right one for you...

4.Public Database Plugin

4.1.About

The Public Database is the one you can use if you aren't able to do a local database. When you enable this plugin all the records driven on your server will be stored on the Aseco server in the internet.

Only requirement is an existing account on www.tmbase.de

4.2.Install

This database is easy to install as well, because you don't have to install any database server on your computer.

Now you will have to create an server with your specific account on www.tmbase.de . To do so, log into TmBase with the account you have created in chapter 3.3c: "Create Account". Now go to your profile and Add a new server to your server management.

Note your server login and password, you will need it again to configure the the plugin. Your server is added to the official TmBase server list now and shown to other players.

It is installed by just enabling the plugin in your plugins.xml.

How to do this is explained in chapter 6 on page 8 (see "Install a new plugin"). The file name of this plugin is "plugin.localdatabase.php" and it's copied into the plugins directory already.

4.3.Configure

Sometimes plugins have their own configuration files, like Aseco itself. These files in general have the extension ".xml" which is a platform independent file format to exchange information.

Open the "publicdatabase.xml" - which is the configuration file for the public database plugin - with a text editor, you will see some text like that:

```
<?xml version="1.0" encoding="utf-8" ?>
<settings>
  <!-- go to www.tmbase.de and create an account before -->
  <!-- Master Server is running at 193.47.83.151:10000 -->
  <server_login>login</server_login>
  <server_password>password</server_password>
  <server_ip>193.47.83.151</server_ip>
  <server_port>10000</server_port>
  <!-- Do you want Aseco to display the records? -->
  <display>true</display>
</settings>
```

publicdatabase.xml

server_login is the IP address or host name where your MySQL server is running (may be "127.0.0.1" or "localhost").

server_login is the login you entered when you created an server with your TmBase account.

server_password is the password matching your mysql_login.

server_ip and **server_port** contain the address of the Aseco Master Server (this only needs to be changed when it's requested by the author).

display sets whether this database is the primary database Aseco should use to display records to players on the server. Databases which are not set to primary will work in background - the user will not notice they are running.

Warning: it may come to a conflict if you run both database plugins as primary databases. Only one of them is allowed to run on primary mode!

5. Local Database

5.1. About

The "plugin.localdatabase.php" which is copied to the plugins folder already, too, is something for users that are a bit experienced with databases. It makes it possible to store records on your own database and prevent them from being mixed up with records from servers all over the world of Trackmania. You can use this database then to fill your website with some statistics, records driven on your server and so on!

In every case you'll need to install a MySQL server which will be used to write records to the database and read out of them.

The database will contain all the records, players, votings etc.

You can download the latest MySQL server here: [download MySQL 5](#)

On Windows computers you will get an easy setup which will guide you through the installation process.

5.2. Install

The MySQL server you've installed now will need a structure to know how to store records, players, votes etc. That's why there's a folder in your Aseco root directory called "localdb". It contains a file named "aseco.sql" which will tell your server how to organize the many information Aseco is going to put into the database.

To import this file go to the command line and browse into a folder called "bin", which is located in your MySQL directory, using the "cd" command:

```
cd "C:\Program Files\MySQL5\bin"
```

Now we are going to log onto the MySQL server:

```
mysql -u login -p
```

Replace "login" with your MySQL login, then you should be asked for your password.

In the last step we create a new database and import the database structure from the file (replace the text written in italic font after your own needs):

```
mysql> CREATE DATABASE databasename
```

```
mysql> USE databasename;
```

```
mysql> source D:\Aseco\localdb\aseco.sql
```

5.3.Configure

As the public database plugin the local one uses its own xml file to be configured. Open the localdatabase.xml which is located in the folder you installed Aseco to:

```
<?xml version="1.0" encoding="utf-8" ?>
<settings>
  <!-- MySQL Server Settings -->
  <mysql_server>127.0.0.1</mysql_server>
  <mysql_login>login</mysql_login>
  <mysql_password>password</mysql_password>
  <mysql_database>databasename</mysql_database>
  <!-- Do you want Aseco to display the records? -->
  <display>>false</display>
</settings>
```

localdatabase.xml

mysql_server is the address your MySql server is running on (probably "127.0.0.1" which is the address of your local computer).

mysql_login the login Aseco should use to connect to the server.

mysql_password has to match the **mysql_login**.

mysql_database is the database you imported the aseco.sql into.

display sets whether this database is the primary database Aseco should use to display records to players on the server. Databases which are not set to primary will work in background - the user will not notice they are running.

Warning: it may come to a conflict if you run both database plugins as primary databases. Only one of them is allowed to run on primary mode!

6.Plugins

You can use PlugIns to customize your version of Aseco. PlugIns in general will add new functionality to your server. All PlugIns are located in the same folder ("plugins"). If you copy a plug-in into that folder it will not be included automatically. You first will have to tell Aseco to load that plug-in, Aseco will read a configuration file ("plugins.xml") in the root folder to know which files to execute.

```
<?xml version="1.0" encoding="utf-8" ?>
<aseco_plugins>
  <plugin>plugin.publicdatabase.php</plugin>
  <plugin>chat.admin.php</plugin>
  <plugin>chat.help.php</plugin>
  <plugin>chat.me.php</plugin>
  <plugin>chat.records.php</plugin>
  <plugin>chat.stats.php</plugin>
  <plugin>chat.vote.php</plugin>
</aseco_plugins>
```

6.1.Install a new plugin

If you want to add a new plugin you have to copy the plugin file into the plugins directory first if it is not copied into it already. Now you have to modify the plugin list ("plugins.xml"). Create a new line above the one which says "</aseco_plugins>". Then you will have to write the file-name of your plugin (i.e.: "plugin.localdatabase.php") and put a "<plugin>" in front and a "</plugin>" behind it.

```
<plugin>plugin.localdatabase.php</plugin>
</aseco_plugins>
```

When you launch Aseco now it should say something like that:

```
[Aseco] Load plugin [plugin.localdatabase.php]
```

6.2.Remove a plugin

There are two ways to delete a plugin.

You can either fully remove it by deleting the file out of the plugins directory and cut the line with the plugin name out of the plugins.xml.

Or you just open the plugins.xml and comment the line which defines the specific plugin.

By putting a "<!--" before and a "-->" behind the plugin you can tell Aseco to jump over that line and resume reading at the next one.

```
<!--<plugin>plugin.localdatabase.php</plugin> -->
</aseco_plugins>
```

7. Customize

7.1. Chat Plugins

You can create your own chat commands to make your server more individual. The chapter Customize requires some to good knowledge in Php. If you are quite beginner or never heard of Php, then it will get hard for you to go on.

Now I just go and try to show you how to do it by an example. First I will create a file in the plugins directory of Aseco („C:\Programs\Aseco\plugins“). After that you will have to add your file in the plugins.xml. Since 0.6.0 only plugins which are written into this file are included.

I start building up the standard structure for a Php file.

```
<?php
?>
```

Otherwise the compiler does not realize this is a Php code and just echo our code in the console.

Next step is to create the function which should be called when a user uses the specified command. In this example we just want to send „Hello“ to all players on the server when a user is using the command „/hello“.

```
<?php
function chat_hello() {
}
?>
```

The function to a chat command always has to be named like „chat_commandname“.

Before this function is called by Aseco or any help is displayed about it when typing „/help“ we have to tell Aseco that there is a chat command named like that.

```
<?php
Aseco::addChatCommand("hello", "sends a hello message to all players");

function chat_hello() {
}
?>
```

Okay, now the command will be displayed in the help.

But how do we send information to the server or where do we know who had sent this command?

Aseco gives two parameters to the function.

```
<?php
Aseco::addChatCommand("hello", "sends a hello message to all players");

function chat_hello($aseco, $command) {
```

```
}  
?>
```

The first parameter is a reference to the Aseco object. You have direct access to all methods and fields.

The second one is an array containing information about who sent the command and which additional parameters were used.

The following chat message

```
„[DerFlo]: /hello param1 param2“
```

will end up in an array like that:

```
$command = array („author“ => Player Object,  
                „params“ => array(0 => „param1“,  
                                 1 => „param2“));
```

We will use the Aseco reference now to access the dedicated server.

```
<?php  
Aseco::addChatCommand("hello", "sends a hello message to all players");  
  
function chat_hello($aseco, $command) {  
    $aseco->addCall („ChatSendServerMessage“, array („Hello“))  
}  
?>
```

7.2.Event oriented Plugins

Here a little example again:

We want to display a message if a user uses bad words like „fuck“.

So we just create a function again, the name doesn't play a role here, so use the name to describe what it's doing, like:

```
<?php  
function checkForBadWords($aseco, $command) {  
}  
?>
```

The next thing is to set an event when our little function should be executed.

```
<?php  
Aseco::registerEvent("onChat", "checkChatForBadWords");  
  
function checkChatForBadWords($aseco, $command) {  
}  
?>
```

with registerEvent we tell Aseco when to call which function.

In this example we told it, that whenever a player wrote something in the chat our little function with the name „checkChatForBadWords“ should be called.

Aseco will use two parameters again when calling the function.

The first is the reference to the Aseco object.

And the second one is the response of the PlayerChat callback the dedicated server send to Aseco.

Its structure looks like that:

```
(int PlayerUid, string Login, string Text, bool IsRegisteredCmd);
```

Now we can use these parameters to react on chat messages.

```
<?php
Aseco::registerEvent("onChat", "checkChatForBadWords");

function checkChatForBadWords($aseco, $command) {
    if ($command[2] == "fuck") {
        $message = "We don't use words like that on this server!";
        $aseco->addCall("ChatSendServerMessage", array($message));
    }
}
?>
```

Further registerable events are:

Name	Description
onChat	A chat message was written. First parameter is a reference to the aseco object. Second parameter: (int PlayerUid, string Login, string Text, bool IsRegisteredCmd);
onStartup	When Aseco initialized and connected to the server. First parameter is a reference to the aseco object. Second parameter is an empty array.
onPlayerConnect	A player connected to the server. First parameter is a reference to the aseco object. Second parameter is a Player Object.
onPlayerDisconnect	A player disconnected from the server. First parameter is a reference to the aseco object. Second parameter is a Player Object.
onPlayerFinish	A player crossed the finish and set up a time. First parameter is a reference to the aseco object. Second parameter is a Record Object.
onCheckpoint	A player crossed a checkpoint. First parameter is a reference to the aseco object. Second parameter: (int PlayerUid, string Login, int Time, int Score, int CheckpointIndex).
onNewChallenge	New challenge has been loaded. First parameter is a reference to the aseco object. Second parameter is a Challenge Object.
onEndRace	Race has ended, players see the rankings now. First parameter is a reference to the aseco object. Second parameter is the result array from the dedicated's EndRace callback.
onPlayerServerMessageAnswer	When player clicks one of the buttons in the window you can open by rpc command. First parameter is a reference to the aseco object. Second parameter is described in the ListCallbacks.html of the dedicated server.
onPlayerWins	A player won a race. First parameter is a reference to the aseco object. Second parameter contains the object of the player.

It would have been too much work to explain the second parameters in all details...

If you should have problems with the content of any, I suggest to use the function **print_r()** to examine the value of this parameters.

8.Chat Documentation

8.1. Public Commands

Name	Description
/help	Displays all chat commands. You can add a specific chat command as parameter to get a description about it. (/help help)
/me	You can use this command to express emotions. It will replace /me by your NickName and attach the parameter as emotion. (/me is driving too fast)
/recs	Displays TOP 5 records on the current track.
/vote x	Votes for a track, replace 'x' with a number from 0 to 10.
/score	Displays the score of the current track.
/admin	Only available to admin, see 5.2
/stats (login)	Show statistics about oneself or a specific player

Next versions will be done to advance chat functions!

8.2. Admin only Commands

Name	Description
/admin setservername	Sets servername to the first parameter attached (/admin setservername ASECO Powerserver)
/admin setmaxplayers	Sets a new maximum of players who can join. (/admin setmaxplayers 17)
/admin nextmap	Skips current track and forces the server to load next challenge.
/admin restartmap	Forces server to restart the curren map.
/admin kick	Kicks a player with the specified login. (/admin kick derflo)

More coming ...

9. Bug Reporting

You found a bug in Aseco?

Then go and report it, our forums are available 24 hours 7 days a week!

<http://forum.tmbase.de>

10.Change Log

Documentation started at version 0.5.3

Version	Changes
0.6.1	<ul style="list-style-type: none">✓ Solved some problems with the Master Server.✓ TMX support, saving Online World records now!✓ Improved database plugins and made them compatible with each other. Can be run parallel now!✓ Fixed some bugs with the public database.
0.6.0	<ul style="list-style-type: none">✓ Created an Aseco Master Server which is the new link between the public database and all Aseco Clients. Each login can only be used for one server since now!✓ Fixed some database and xml bugs which produced a warning like „No records this round!“✓ Improved the local database plugin. Now there is a players table.✓ Fixed admin chat commands.✓ New feature: /stats command.✓ 5. world record is saved and read from database now.
0.5.4	<ul style="list-style-type: none">✓ New XML parser for better performance.✓ Fixed chat commands /score, /me and /vote.
0.5.3	<ul style="list-style-type: none">✓ New object type „Server“.✓ Getting game type from dedicated server now.✓ Chat commands /help and /recs were fixed for TMS and TMO.✓ fetching all news from data server.✓ waiting for server when Aseco starts, but server hasn't loaded

Version	Changes
	a track yet. ✓ Modified the formatText(), console() and console_text(). Built in a logfile (but does not log errors yet.)
0.5.0, 0.5.1, 0.5.2	✓ No documentations.

11.Team

11.1.Aseco

Florian Schnell

11.2.Website

Sascha Willecke

... is still hard working to give you stats about players and servers!

11.3.Plugins

Iain Surgey

... famous creator of RASP 0.2 which you can find here:

<http://www.tm-forum.com>

