

ShopNC B2B2C 商城安装手册

天津市网城天创科技有限责任公司

2014-年 2 月

1. 系统介绍

ShopNC B2B2C 商城是 ShopNC 于 2014 年最新推出，采用 PHP5+MySQL 技术为基础，采用 OOP（面向对象）方式进行核心框架搭建，结合 MVC 模式进行开发的一套电商平台系统。

2. 运行环境

项目	说明
操作系统	推荐使用 Linux，兼容 Windows，内存 >= 16G，硬盘 >= 300G
Web 服务器	Apache 或 Nginx
数据库	MySQL5 及以上版本
PHP	PHP5.2 及以上版本，支持以下扩展： php_mysql php_mysqli php_gd2 php_mbstring 支持以下函数： mysql_connect, fsockopen, gethostbyname, file_get_contents, mb_convert_encoding, json_encode 其它项： file_uploads = On upload_max_filesize = 1M

ShopNC 商城系统运行环境

推荐系统运行于 Linux + Nginx 环境。

3. 安装详解

系统运行的最佳环境是 Linux + Nginx，下面介绍一下此环境的搭建过程。

3.1 安装 Nginx

主配置文件：/Data/app/nginx/conf/nginx.conf

服务控制：service nginx (start|stop|restart|reload|configtest)

核心命令：

```
wget http://nginx.org/download/nginx-1.4.2.tar.gz

tar zxvf nginx-1.4.2.tar.gz

cd nginx-1.4.2

./configure \

    "--prefix=/Data/app/nginx" \

    "--sbin-path=/Data/app/nginx/sbin/nginx" \

    "--conf-path=/Data/app/nginx/conf/nginx.conf" \

    "--error-log-path=/Data/app/nginx/logs/error.log" \

    "--http-log-path=/Data/app/nginx/logs/access.log" \

    "--pid-path=/Data/app/nginx/var/nginx.pid" \

    "--lock-path=/Data/app/nginx/var/nginx.lock" \

    "--http-client-body-temp-path=/tmp/clientbody" \

    "--http-proxy-temp-path=/tmp/proxy" \

    "--http-fastcgi-temp-path=/tmp/fastcgi" \

    "--http-uwsgi-temp-path=/tmp/uwsgi" \

    "--http-scgi-temp-path=/tmp/scgi" \

    "--user=www" \

    "--group=www" \

    "--without-select_module" \

    "--without-poll_module" \

    "--with-file-aio" \

    "--with-http_realip_module" \

    "--with-http_ssl_module" \

    "--with-openssl=/Data/software/openssl" \

    "--with-http_gzip_static_module" \

    "--with-zlib=/Data/software/zlib" \

    "--with-http_stub_status_module" \

    "--without-http_ssi_module" \
```

```
"--without-http_userid_module" \  
"--without-http_geo_module" \  
"--without-http_empty_gif_module" \  
"--without-http_map_module" \  
"--without-mail_pop3_module" \  
"--without-mail_imap_module" \  
"--without-mail_smtp_module" \  
"--with-pcre=/Data/software/pcre"
```

```
make && make install
```

编辑配置文件

```
vi /Data/app/nginx/conf/nginx.conf
```

加入如下内容

```
user www www;  
  
error_log /Data/app/nginx/logs/error.log error;  
  
pid /Data/app/nginx/var/run/nginx.pid;  
  
lock_file /Data/app/nginx/var/lock/nginx;  
  
worker_rlimit_nofile 102400;  
  
worker_processes 2;  
  
#worker_cpu_affinity 00000100 00001000 00010000 00100000;  
  
events {  
    worker_connections 10240;  
    use epoll;  
    multi_accept on;  
}
```

```
http {  
  
#core  
  
    include          mime.types;  
  
    default_type     application/octet-stream;  
  
    client_body_buffer_size      8k;  
  
    client_body_temp_path        /tmp/client_body_temp;  
  
    client_body_timeout          30;  
  
    client_header_buffer_size    32k;  
  
    large_client_header_buffers 4 32k;  
  
    client_header_timeout        30;  
  
    client_max_body_size         32m;  
  
    #keepalive_disable  msie6 safari;  
  
    keepalive_timeout  3;  
  
    tcp_nodelay        on;  
  
    send_timeout 30;  
  
    sendfile    on;  
  
    tcp_nopush on;  
  
    server_names_hash_max_size    512;  
  
    server_names_hash_bucket_size 128;  
  
    server_tokens off;  
  
    open_file_cache off;  
  
#index  
  
    index  index.php index.html index.htm;  
  
#fastcgi  
  
    fastcgi_connect_timeout    60;  
  
    fastcgi_read_timeout       60;  
  
    fastcgi_send_timeout       60;
```

```
fastcgi_temp_path          /tmp/fastcgi_temp 1 2;

fastcgi_buffer_size        64k;

fastcgi_buffers            256 4k;

fastcgi_max_temp_file_size 256k;

fastcgi_intercept_errors   on;

fastcgi_index              index.php;

#proxy

proxy_temp_path           /tmp/proxy_temp;

proxy_buffer_size         4k;

proxy_buffering           on;

proxy_buffers             256 4k;

proxy_busy_buffers_size   8k;

#gzip

gzip                      on;

gzip_buffers              16 4k;

gzip_comp_level           1;

gzip_http_version         1.1;

gzip_min_length           1024;

gzip_types                text/css text/xml text/plain text/vnd.wap.wml application/x-javascript application/rss+xml
application/xhtml+xml;

#log module

log_format main '$remote_addr - $remote_user [$time_local] $request '
                '$status' $body_bytes_sent "$http_referer" '
                '$http_user_agent' "$http_x_forwarded_for";

#virtualhost

include vhosts/*;

}
```

保存退出

创建日志与启动服务

```
mkdir -p /Data/app/nginx/logs/www.shopnc.net

chmod 755 /etc/init.d/nginx

chkconfig --add nginx

chown -R www:www /Data/app/nginx/logs/

chown -R www:www /Data/code/

service nginx configtest
```

3.2 安装 PHP

PHP 配置文件：/Data/app/php/etc/php.ini

PHP-FPM 配置文件：/Data/app/php/etc/php-fpm.conf

扩展模块目录：/Data/app/php/ext

服务控制：service php-fpm (start|stop|restart|reload)

核心命令：

安装所需库

```
yum install gd-devel

yum install curl-devel

yum install pcre-devel

yum install zlib-devel

yum install libxml2-devel

yum install bzip2-devel

yum install libxpm-devel

yum install libXpm

yum install libXpm-devel.i386

yum install libXpm-devel.i586

yum install libXpm-devel.i686

yum install mbstring

yum install exif

yum install libicu-devel
```

```
yum install libmcrypt-devel

yum install php-mcrypt

wget
http://downloads.sourceforge.net/project/mcrypt/Libmcrypt/2.5.8/libmcrypt-2.5.8.tar.gz?r=http%3A%2F%2Fsourceforge.net%2Fprojects%2Fmcrypt%2Ffiles%2FLibmcrypt%2F2.5.8%2F&ts=1378526207&use_mirror=jaist

tar xzvf libmcrypt-2.5.8.tar.gz

cd libmcrypt-2.5.8

./configure --prefix=/Data/app/libmcrypt

make && make install
```

编译安装 PHP

切换到 PHP 安装包目录

```
tar xzvf php-5.4.19.tar.gz

cd php-5.4.19

./configure --prefix=/Data/app/php --with-config-file-path=/Data/app/php/etc --with-mysql=/Data/app/mysql --with-libxml-dir --with-pcre-dir --with-mcrypt=/Data/app/libmcrypt --with-gd --with-jpeg-dir --with-png-dir --with-xpm-dir --with-gettext --with-freetype-dir --with-bz2 --with-iconv-dir --with-zlib --with-zlib-dir --with-curl --enable-fpm --with-fpm-user=www --with-fpm-group=www --enable-intl --enable-sockets --enable-gd-native-ttf --enable-gd-jis-conv --enable-ftp --enable-zip --enable-calendar --enable-mbstring --enable-exif --enable-fd-setsize=4096 --disable-short-tags --disable-ipv6

make

make install

cd /Data/software/

cd php-5.4.19
```

安装 mysqli 扩展

```
cd modules/ext/mysqli/

/Data/app/php/bin/phpize

./configure --with-mysqli=/Data/app/mysql/bin/mysql_config --with-php-config=/Data/app/php/bin/php-config

make
```



```
cp modules/mysql.so /Data/app/php/ext/
```

编辑 php.ini 文件, 增加以下内容

```
vi /Data/app/php/etc/php.ini
```

增加 mysql 扩展文件所在路径

```
extension = "/Data/app/php/ext/mysql.so"
```

```
mysql.max_persistent = -1
```

保存退出

注意: file_uploads = On, 即上传功能需要开启

将 php-fpm 加入系统服务, 并开机启动

```
chmod 755 /etc/init.d/php-fpm
```

```
chkconfig --add php-fpm
```

```
chkconfig --level 35 php-fpm on
```

```
service php-fpm start
```

3.3 安装 MySQL

主配置文件: /etc/my.cnf

数据库目录: /Data/app/mysql/data

shopnc 数据库信息

用户名: shopnc

密码: xxxxxxxx

服务器控制: service mysqld (start|stop|restart)

核心命令:

当前目录位于 /Data/

```
tar xzvf mysql-5.5.33-linux2.6-x86_64.tar.gz
```

```
mv mysql-5.5.33-linux2.6-x86_64 app/
```

```
cd app/
```

指定运行账户和数据存放目录

```
useradd mysql
```

```
chown -R mysql:mysql ./mysql-5.5.33-linux2.6-x86_64/
```

```
chown -R mysql:mysql ./mysql-5.5.33-linux2.6-x86_64
```

```
ln -s /Data/app/mysql-5.5.33-linux2.6-x86_64 /Data/app/mysql
```

```
cd mysql
```

编辑配置文件 my.cnf

```
vi /etc/my.cnf
```

加入以下内容

```
[client]
```

```
port = 3306
```

```
socket = /tmp/mysql.sock
```

```
[mysqld]
```

```
port = 3306
```

```
socket = /tmp/mysql.sock
```

```
basedir = /Data/app/mysql
```

```
datadir = /Data/app/mysql/data
```

```
#bind-address = 0.0.0.0
```

```
character-set-server = utf8
```

```
default-storage-engine = InnoDB
```

```
skip-external-locking
```

```
skip-name-resolve
```

```
#skip-networking
```

```
back_log = 300
```

```
key_buffer = 64M
```

```
max_allowed_packet = 16M
```

```
table_open_cache = 8192
```

```
max_connections = 1024
```

```
max_connect_errors = 100
```

```
sort_buffer_size      = 16M
read_buffer_size      = 16M
read_rnd_buffer_size  = 8M
thread_cache_size     = 16
query_cache_size      = 32M

log_error              = /Data/app/mysql/log/error.log
slow_query_log        = ON
slow_query_log_file   = /Data/app/mysql/log/slow.log
long_query_time       = 2
log-queries-not-using-indexes
thread_concurrency    = 8

server-id = 1

log-bin               = mysql-bin-master
binlog_format        = mixed
sync_binlog          = 1
#expire-logs-days    = 7
max_binlog_size      = 512M

#### InnoDB #####

innodb_file_per_table
innodb_data_home_dir = /Data/app/mysql/data
innodb_data_file_path = ibdata1:32M:autoextend
innodb_autoextend_increment = 8
innodb_buffer_pool_size = 2G
innodb_additional_mem_pool_size = 32M
```

```
innodb_log_group_home_dir      = /Data/app/mysql/data
innodb_log_files_in_group      = 3
innodb_log_file_size           = 64M
innodb_log_buffer_size         = 8M
innodb_write_io_threads        = 8
innodb_read_io_threads         = 8
innodb_thread_concurrency      = 16
innodb_flush_method            = O_DSYNC
innodb_flush_log_at_trx_commit = 1
innodb_max_dirty_pages_pct     = 90
innodb_lock_wait_timeout       = 120

#innodb_force_recovery         = 1

#innodb_fast_shutdown

[mysqldump]

quick

max_allowed_packet             = 16M

[mysql]

default-character-set          = utf8

no-auto-rehash

[myisamchk]

key_buffer                     = 256M
sort_buffer_size               = 256M
read_buffer                    = 2M
write_buffer                   = 2M
```

保存退出

然后安装初始数据库

```
cd /Data/app/mysql/  
  
./scripts/mysql_install_db --user=mysql  
  
cd data/  
  
cd ..
```

加到系统服务并开机启动

```
cp support-files/mysql.server /etc/init.d/mysqld  
  
chmod 755 /etc/init.d/mysqld  
  
chkconfig --add mysqld  
  
chkconfig --level 35 mysqld on
```

设置日志

```
cd /Data/app/mysql  
  
mkdir log  
  
chown mysql:mysql log/  
  
service mysqld start
```

3.4 安装商城程序

在 Nginx、PHP、MySQL 安装并配置完成以后，可以安装 ShopNC B2B2C 商城了。

首先需要仔细阅读安装协议

ShopNC【B2B2C】电商系统安装协议

感谢您选择ShopNC【B2B2C】电商系统。本系统是天津市网城天创科技有限责任公司自主开发、独立拥有版权、集资讯、商城、讨论组、微商城、在线IM、客户管理于一体的门户型解决方案。官方网址为 <http://www.shopnc.net>，官方论坛网址为 <http://bbs.shopnc.net>。

用户须知：本协议是您与网城天创公司之间关于您安装使用网城天创公司提供的ShopNC【B2B2C】电商系统及服务的法律协议。无论您是个人或组织、盈利与否、用途如何（包括以学习和研究为目的），均需仔细阅读本协议。请您审阅并接受或不接受本协议条款。如您不同意本协议条款或网城天创公司随时对其的修改，您应不使用或主动取消网城天创公司提供的ShopNC【B2B2C】电商系统。否则，您的任何对ShopNC【B2B2C】电商系统使用的行为将被视为您对本服务条款全部的完全接受，包括接受网城天创对服务条款随时所做的任何修改。

本服务条款一旦发生变更，网城天创公司将在网页上公布修改内容。修改后的服务条款一旦在网页上公布即有效代替原来的服务条款。如果您选择接受本条款，即表示您同意接受协议各项条件的约束。如果您不同意本服务条款，则不能获得使用本服务的权利。您若有违反本条款规定，网城天创公司有权随时中止或终止您对ShopNC【B2B2C】电商系统的使用资格并保留追究相关法律责任的权利。

在理解、同意、并遵守本协议的全部条款后，方可开始使用ShopNC【B2B2C】电商系统。您可能与网城天创公司直接签订另一书面协议，以补充或者取代本协议的全部或者任何部分。

网城天创拥有本软件的全部知识产权。本软件只供许可协议，并非出售。网城天创只允许您在遵守本协议各项条款的情况下复制、下载、安装、使用或者以其他方式受益于本软件的功能或者知识产权。

同意协议进入安装

不同意

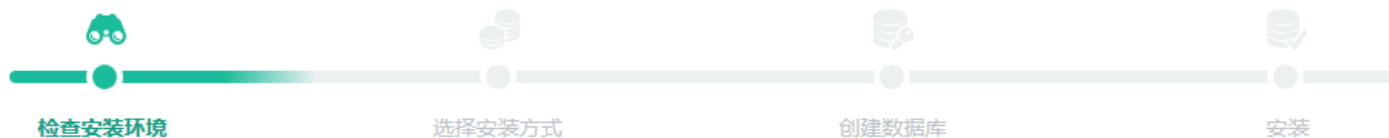
Powered by ShopNC【B2B2C】

版权所有 2007-2014 © 天津市网城天创科技有限责任公司
软件注册登记编号：软著登字第0650314号

阅读并同意后开始检查安装环境，如果检测有未通过项，将显示红色提示并无法进行下一步

Step.1 开始安装

检测服务器环境及文件目录权限



环境检查

项目	ShopNC所需	最佳配置推荐	当前服务器
操作系统	无限制	linux	✔ Linux
PHP版本	5.2	5.3	✔ 5.4.24
GD库	2.0	2.0	✔ 2.1.0
附件上传	未限制	2M	✔ 2M
磁盘空间	100M	>100M	✔ 44467M

目录、文件权限检查

目录文件	所需状态	当前状态
data/cache	可写	✘ 不可写
data/cache/adv	可写	✘ 不可写
data/cache/rec_position	可写	✘ 不可写
data/config	可写	✘ 不可写
data/log	可写	✘ 不可写

待所有检查项均通过后，进入下一步选择安装方式（目前只有一个选项）

Step.2 选择安装方式

根据需求选择系统模块完全或手动安装



完全安装 ShopNC [B2B2C] 电商系统

[上一步](#)
[下一步](#)

下一步输入数据库账号等信息



检查安装环境



选择安装方式



创建数据库



安装

数据库信息

数据库服务器	<input type="text" value="localhost"/>	数据库服务器地址，一般为localhost
数据库名	<input type="text" value="shopnc"/>	
数据库用户名	<input type="text" value="root"/>	
数据库密码	<input type="password" value="....."/>	
数据库表前缀	<input type="text" value="shopnc_"/>	同一数据库运行多个ShopNC程序时，请修改前缀
数据库端口	<input type="text" value="3306"/>	数据库默认端口一般为3306

演示数据，如果选择安装演示数据，请下载[图片文件压缩包](#)

网站信息

站点名称	<input type="text" value="ShopNC B2B2C商城 - 演示站"/>	输入站点名称，安装后可在平台设置中进行修改
管理员帐号	<input type="text" value="admin"/>	
管理员密码	<input type="password" value="....."/>	管理员密码不少于6个字符
重复密码	<input type="password" value="..... "/>	确保两次输入的密码一致

如果选择安装演示数据，请从官方下载图片压缩包，解压到商城根目录。

下一步开始安装数据库

Step.4 安装数据库

正在执行数据库安装



数据表 shopnc_circle_thpolloption ... 创建成功
 数据表 shopnc_circle_thpollvoter ... 创建成功
 数据表 shopnc_circle_threply ... 创建成功
 数据表 shopnc_micro_adv ... 创建成功
 数据表 shopnc_micro_comment ... 创建成功
 数据表 shopnc_micro_goods ... 创建成功
 数据表 shopnc_micro_goods_class ... 创建成功
 数据表 shopnc_micro_goods_relation ... 创建成功
 数据表 shopnc_micro_like ... 创建成功
 数据表 shopnc_micro_member_info ... 创建成功
 数据表 shopnc_micro_personal ... 创建成功
 数据表 shopnc_micro_personal_class ... 创建成功
 数据表 shopnc_micro_store ... 创建成功
 初始化数据 ... 成功

安装完成，下一步...

安装数据库成功，下一步提示安装成功并显示安装相关信息

 **程序已成功安装**
 选择您要进入的页面

				
商城 线上购物、开店、交易...	资讯 CMS资讯、画报、专题...	圈子 主题、网友、商品...	微商城 随心看、个人秀、店铺街	系统管理 电商系统后台

系统管理默认地址: http://192.168.1.201/v2014_test2/admin
 网站首页默认地址: http://192.168.1.201/v2014_test2
 如选择安装了演示数据，网站默认会员帐号和密码均为shopnc。
 网站默认商家帐号: shopnc_seller; 密码: shopnc。请下载[图片文件压缩包](#)

安装完成后，进入后台 » 设置 » 清理缓存 » 清理全部缓存。

ShopNC

[控制台](#) | [设置](#) | [商品](#) | [店铺](#) | [会员](#) | [交易](#) | [网站](#) | [运营](#) | [统计](#) | [微商城](#) | [CMS](#) | [圈子](#)

ADMIN CONTROL PANEL

您的位置: [设置](#) > [清理缓存](#)

- 站点设置
- 账号同步
- 上传设置
- SEO设置
- 消息通知
- 支付方式
- 权限设置
- 快递公司
- 配送地区
- 商品设置
- 清理缓存
- 性能优化
- 搜索设置
- 操作日志

清理缓存

全部

商城配置

商品分类

广告缓存

团购缓存(地区|分类|价格区间)

底部导航

首页

表结构

SEO

快递公司

店铺分类

店铺等级

圈子成员等级

点击后台右上解的商城首页链接，查看一下商城首页，内容正确展示，至此安装过程顺利完成。

您好，欢迎来到 ShopNC B2B2C商城 - 演示站 [登录] [注册] [商家管理中心](#) 我的订单 ▾ 我的收藏 ▾ 客户服务 ▾

ShopNC

商城

[我的商城 ▾](#)

精确定位客户市场 专注电商系统服务 热门搜索: 周大福 内衣 金史密斯 手机

所有商品分类 ▾
首页
团购
品牌
积分中心
门户
圈子
微商城

- + 服饰鞋帽
T恤 衬衫 针织衫 雪纺衫
- + 礼品箱包
钱包卡包 手拿包 单肩包
- + 家居家装
床品件套 被子 枕芯枕套
- + 数码办公
手机电池 蓝牙耳机
- + 家用电器
平板电视 空调 冰箱 洗衣机
- + 个人护肤
洁面乳 爽肤水 精华露
- + 珠宝首饰
项链 手链脚链 戒指 耳饰
- + 食品饮料
饼干蛋糕 糖果巧克力

清仓

年末特惠

满99元美丽故事
终身包邮卡 送

佳节大献礼
茶满中秋

佳节倍思亲
品茶好时节

孩子喜欢
遥控悍马

宝贝喜欢玩
父母也喜欢

天气凉了
情侣家居服

秋意浓爱意浓
家居服也情侣

七天包退 正品保障 闪电发货

近期团购

东阿阿胶桃花姬阿胶糕300g, 芝麻...

¥95.00

立即团

已售0 262天10小时 43分57秒

商城公告

- 火爆销售中 (2013-02-28)
- 功能使用说明 (2011-12-02)
- 提示信息 (2011-07-10)
- 如何扩充水印字体库 (2011-07-09)
- 管理功能说明 (2011-07-09)

热卖商品

疯狂抢购

猜您喜欢

热评商品

新品上架

限时折扣

3.5 设置定时任务

系统中的一些操作（如生成统计数据、生成搜索索引数据等）需要使用系统任务定时触发来完成。待触发的程

序文件位于商城目录下的 data/crontab 目录中 ,index.php 是统一入口文件 ,include 目录中的文件来完成各个子功能。

定时任务触发目录及触发频率参考如下：

```
# /usr/local/php/bin/php 为 php 可执行文件所在路径
# 更新商品促销到期状态，建议执行频率：1 天
/usr/local/php/bin/php /wwwroot/data/crontab/index.php goods promotion

# 执行通用任务，建议执行频率：1 天
/usr/local/php/bin/php /wwwroot/data/crontab/index.php goods common

# 更新首页商品价格，建议执行频率：10 分钟
/usr/local/php/bin/php /wwwroot/data/crontab/index.php goods web_update

# 会员相关数据统计，建议执行频率：1 天
/usr/local/php/bin/php /wwwroot/data/crontab/index.php stat memberstat

# 生成结算单，建议执行频率：每月 1 号(1:00 ~ 4:00) 执行
/usr/local/php/bin/php /wwwroot/data/crontab/index.php order create_bill

# 全量创建索引，初始安装时，执行一次即可
/usr/local/php/bin/php /wwwroot/data/crontab/index.php xs create

# 更新增量索引，建议执行频率：5 小时
/usr/local/php/bin/php /wwwroot/data/crontab/index.php xs update
```

示例：

```
# 每天 1:00 执行
0 1 * * * /usr/local/php/bin/php /wwwroot/data/crontab/index.php goods promotion

# 每天 1:30 执行
30 1 * * * /usr/local/php/bin/php /wwwroot/data/crontab/index.php goods common

# 每 10 分钟执行一次
*/10 * * * * /usr/local/php/bin/php /wwwroot/data/crontab/index.php goods web_update

# 每天 2:00 执行
```

```
0 2 * * * /usr/local/php/bin/php /wwwroot/data/crontab/index.php stat memberstat

# 每月1号3:00执行
0 3 1 * * /usr/local/php/bin/php /wwwroot/data/crontab/index.php order create_bill

# 每5小时执行一次
*/5 * * * * /usr/local/php/bin/php /wwwroot/data/crontab/index.php xs update
```

3.6 开启高级缓存

3.6.1 安装缓存扩展

系统默认只是缓存了系统基本配置信息（即后台清理缓存工具涉及的内容），缓存信息如下图：

- | | | | | |
|-------------------------------|---------------------------------|-------------------------------|---|-------------------------------|
| <input type="checkbox"/> 商城配置 | <input type="checkbox"/> 商品分类 | <input type="checkbox"/> 广告缓存 | <input type="checkbox"/> 团购缓存（地区 分类 价格区间） | <input type="checkbox"/> 底部导航 |
| <input type="checkbox"/> 首页 | <input type="checkbox"/> 表结构 | <input type="checkbox"/> SEO | <input type="checkbox"/> 快递公司 | <input type="checkbox"/> 店铺分类 |
| <input type="checkbox"/> 店铺等级 | <input type="checkbox"/> 圈子成员等级 | | | |

其它数据均未使用缓存，这样造成了系统性能下降。系统强烈建议开启非文件缓存（redis、memcache、xcache、apc、eaccelerator），建议优先使用 redis 或 memcache。开启非文件缓存后，系统将会自动缓存店铺、会员、商品、统计类、搜索结果等局部数据信息，使用系统性能得到较大提高。开启非文件缓存首先确保服务器上相应缓存软件安装正确。

Redis 服务器安装

```
yum install redis

# 如果没有可用的 redis 包，可以先安装 EPEL 后再执行。如 64 位系统，可以先执行
rpm -ivh http://dl.fedoraproject.org/pub/epel/6/x86\_64/epel-release-6-8.noarch.rpm

# 安装完成后动服务器端
/usr/sbin/redis-server /etc/redis.conf
```

Php redis 扩展安装

```
wget https://github.com/nicolasff/phpredis/archive/master.zip

unzip msater

cd phpredis-master/

/usr/bin/phpize
```

```
./configure
make && make install

# 编辑 php.ini , 追加以下内容
extension=/usr/lib64/php/modules/redis.so

# 重启 php-fpm 服务
service php-fpm restart
```

3.6.2 商城配置

缓存服务器与 PHP 扩展安装完成后, 需更改 CONFIG(data/config/cofnig.ini.php)中[cache][save_type]

项, 以使用 redis 为例, CONFIG 配置如下:

```
# 缓存存储类型 支持类型为 file 文件缓存(默认)、redis、memcache、xcache、apc、eaccelerator
$config['cache']['type'] = 'redis';
```

如果缓存类型设置为 memcache 或 redis, 还需要配置下面的 memcache 或 redis 相关参数才会生效

```
# memcached 设置, 可配置多台 memcached 服务器
$config['memcache']['prefix'] = 'nc_'; // memcached 存取时值时, 自动添加的前缀

# 第 1 台 memcached 配置
$config['memcache'][1]['port'] = 11211; // memcached 监听端口号
$config['memcache'][1]['host'] = '192.168.1.13'; // memcached 服务器 IP 地址
$config['memcache'][1]['pconnect'] = 0; //是否长连接 默认 0(否), 1 表示使用长连接

# 第 2 台 memcached 配置
$config['memcache'][2]['port'] = 11211;
$config['memcache'][2]['host'] = '192.168.1.14';
$config['memcache'][2]['pconnect'] = 0;

# 第 N 台 memcached 配置
.....

# redis 配置, 支持一主或一主一从配置
$config['redis']['prefix'] = 'nc_'; // redis 存取时值时, 自动添加的前缀

# redis 主服务器配置
$config['redis']['master']['port'] = 6379;
```

```

$config['redis']['master']['host']      = '192.168.1.15';
$config['redis']['master']['pconnect']  = 0;

# 不启用 redis 从服务器
$config['redis']['slave']               = array();

# 启用 redis 从服务器配置
$config['redis']['slave']['port']       = 6379;
$config['redis']['slave']['host']       = '192.168.1.16';
$config['redis']['slave']['pconnect']   = 0;

```

在开启非文件缓存后，后台性能优化中提供了一键清理操作，可以手动点击清理缓存内容。如图：

性能优化 **内存优化**

当前内存工作状态

内存类型	PHP 扩展环境	Config 设置	清理
Memcache	YES	NO	--
eAccelerator	NO	NO	--
APC	NO	NO	--
Xcache	NO	NO	--
Redis	YES	YES	清理

3.7 开启即时通讯

系统 IM 模块默认并未开启，若想启用 IM 模块，需要先配置 IM 服务器。可以从官网 (<http://nodejs.org/download/>) 下载二进制包解压下载使用，也可以下载源码包编译安装，以下为源码编译过程：

```

wget http://nodejs.org/dist/v0.10.25/node-v0.10.25.tar.gz
tar zxf node-v0.10.25.tar.gz
cd node-v0.10.25

```

```
./configure  
make && make install
```

将安装包内 im 文件夹，放置到非 WEB 目录，然后编辑 im/config.js 文件，配置示例如下：

```
var config = {}; //数据库帐号设置  
  
config['host'] = '10.10.10.50'; //数据库地址  
  
config['port'] = '3306'; //数据库端口  
  
config['user'] = 'shopnc'; //数据库用户名  
  
config['password'] = 'xxxxxx'; //数据库密码  
  
config['database'] = 'shopnc'; //mysql 数据库名  
  
config['tablepre'] = 'shopnc_'; //表前缀  
  
config['insecureAuth'] = true; //兼容低版本  
  
config['debug'] = false; //默认 false  
  
exports.hostname = ''; //授权连接的域名或 IP, 为空不限制  
  
exports.port = 8090; //服务器所用端口号, 默认 8090, 也可自定义  
  
exports.config = config;
```

然后启动 node (假设 IM 文件包目录为 /data/im/)

```
/usr/local/bin/node /data/im/chat.js
```

如果出现以下内容说明配置成功：

```
info - socket.io started  
  
mysql connected
```

然后停止当前 node，将 node 以后台方式启动：

```
nohup /usr/local/bin/node /data/im/chat.js >> /output.log &
```

然后编辑商城根目录 chat/config/config.ini.php，示例如下：

```
define('NODE_SITE_URL', 'http://www.shopnc.net:8090');  
  
define('CHAT_SITE_URL', 'http://www.shopnc.net/chat'); # 这里假设 chat 位于 WEB 根目录
```

最后编辑商城根目录 shop/templates/default/footer.php

去掉以下行的 PHP 注释

```
<?php //echo getChat($layout);?>
```

使用会员账号登录到商城，如果发现右下角出现 IM 图标，说明 IM 配置成功了。如下图：



启动成功后，将 node 启动命令加入到系统启动中。

如果右下角未出现 IM 框，请检查防火墙中是否禁止了 node 监听端口。

3.8 开启全文搜索

系统在默认配置下，前台商品搜索采用时时查询数据库的方式处理，在大数据高并发下，这样会对数据库造成较大压力，建议开启全文搜索。开启后系统会定时将商品的信息索引到搜索服务器（只索引前台搜索排序相关的字段），当前台搜索某关键词（或分类、品牌、属性）时，系统会跟据索引查找搜索服务器，然后返回商品主键 ID（返回数量为每页显示数），最后系统将这些商品主键作为条件查询数据库，显示查询结果。

3.8.1 安装配置

系统默认使用 xunsearch 作为全文搜索引擎，登录迅搜索（<http://www.xunsearch.com/>）官网并下载安装最新的稳定安装包，安装包内含安装说明，这里不再赘述。

安装完成后，更改 CONFIG(data/config/config.ini.php)文件全文搜索部分，配置如下：

```
# 全文检索配置，支持 true(开启) 或 false(关闭)，全文检索的详细配置，请参考本帮助的全文检索章节  
  
# 需编辑 data\api\xs\app\shopnc.ini，将 server.index 和 server.search 值修改成自己的 IP 和端口  
$config['fullindexer']['open'] = true;
```



```
# 全文检索配置文件名(默认为 shopnc , 不需要更改)
$config['fullindexer']['appname'] = 'shopnc';
```

然后更改全文搜索配置文件 (data/api/xs/app/shopnc.ini), 将 server.index 和 server.search 值修改成自己的 IP 和端口

shopnc.ini 文件完整解释如下 :

```
# 项目名称 ( 不需要更改 )
project.name = shopnc

# 默认字符集 ( 该项要与当前商城字符集一致 )
project.default_charset = UTF-8

# 索引服务端配置, 默认端口 8383
server.index = 127.0.0.1:8383

# 搜索服务端配置, 默认端口 8384
server.search = 127.0.0.1:8384

# 主键
[pk]
type = id

# 商品 ID
[goods_id]
type = numeric

# 商品名称
[goods_name]
type = title
index = mixed
weight = 5

# 分类名称
[gc_name]
index = mixed
```

```
weight = 3
```

```
# 品牌名称
```

```
[brand_name]
```

```
index = mixed
```

```
weight = 4
```

```
# 商品所在一级分类
```

```
[cate_1]
```

```
index = self
```

```
tokenizer = full
```

```
# 商品所在二级分类
```

```
[cate_2]
```

```
index = self
```

```
tokenizer = full
```

```
# 商品所在三级分类
```

```
[cate_3]
```

```
index = self
```

```
tokenizer = full
```

```
# 品牌 ID
```

```
[brand_id]
```

```
index = self
```

```
tokenizer = full
```

```
# 属性 ID
```

```
[attr_id]
```

```
index = self
```

```
tokenizer = split(_)
```

```
# 店铺 ID
```

```
[store_id]
```

```
index = self
```

```
tokenizer = full
```

```
# 一级地区 ID
[area_id]
index = self
tokenizer = full

# 商品价格 (排序)
[goods_price]
type = numeric

# 商品点击量 (排序)
[goods_click]
type = numeric

# 商品销量 (销量)
[goods_salenum]
type = numeric

# 商品分类 ID
[gc_id]
```

配置完成后，启动 xunsearch 进程、索引商品内容到搜索服务器。

```
# 启动 xunsearch 进程
usr/local/xunsearch/bin/xs-ctl.sh -b inet start

# 清空索引
/usr/bin/php /wwwroot/html/crontab/index.php xs clear

# 全量更新索引
/usr/bin/php /wwwroot/html/crontab/index.php xs create

# 增量更新索引，此命令可以加入系统任务计划中，定时执行，建议执行周期为 1 小时
/usr/bin/php /wwwroot/html/crontab/index.php xs update
```

至此，配置工作已经完成，前台可以体验全文搜索了。

3.8.2 自定义词库

系统使用的索引分词器默认为 scws,当内置的词库不能满足您的应用时，可以使用下面介绍的自定义词库功

能，涉及的相关文件如下（假设 \$prefix 是您的安装目录）

\$prefix/etc/dict_user.txt

打开上述文件，文件开头已经有了相关注释和说明。这个文件就是自定义词库文件，如果您同一台机器上装有多个 xunsearch 服务端，那么每个服务端独自使用自己的自定义词典文件。

添加删除修改自定义词库只要编辑该文件即可，以下为相关规范：

- 文件为纯文本文件，编码必须是 UTF-8，可用任何编辑器修改
- 每行一条记录表示一个词，每行包含 1~4 个字段，字段之间用空格或制表符(\t)分隔
- 字段含义依次表示 “词语”，“词频(TF)”，“逆词频率(IDF)”，“词性(ATTR)”
- 后面三个字段如果省略依次使用 scws 的默认值
- 特殊词性！可用于表示删除该词
- 自定义词典优先于内置词典加载和使用，以 # 开头的行为注释

3.9 开启伪静态

系统支持动态网址和伪静态两种 URL 形式，CONFIG(data/config/config.ini.php)中，url_model 为 true 表示开启伪静态，url_mode 为 false 表示使用默认动态网址形式。系统伪静态是基于 PATHINFO 模式实现的，所以若开启伪静态，除更改 CONFIG 外，还需要配置 APACHE/NGINX 支持 PATHINFO 模式。

注意：当开启伪静态时，CONFIG 中的各子模块 URL 网址必须全部配置并启用。

```
# apache 配置
# 编辑 httpd.conf，加载 mod_rewrite.so
# 编辑 httpd.conf，让 Apache 支持.htaccess
AllowOverride None 改为 AllowOverride All
# 在商城目录下新建.htaccess 文件，
# 若以 php-cgi 运行，加入以下内容
<IfModule mod_rewrite.c>
RewriteEngine on
RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME} !-f
RewriteRule ^(.*)$ index.php [L,E=PATH_INFO:$1]
</IfModule>

# 若以 php5apache2_2.dll 方式运行，加入以下内容
<IfModule mod_rewrite.c>
RewriteEngine on
```

```

RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME} !-f
RewriteRule ^(.*)$ index.php/$1 [QSA,PT,L]
</IfModule>

# nginx 配置
location / {
    if (!-e $request_filename)
    {
        rewrite ^/shop/(.*)$ /shop/index.php?$1;
    }
}

location ~ /\.php$ {
    fastcgi_param PATH_INFO $request_uri;
}

```

3.10 设置域名

系统默认安装后，商城、CMS、圈子、微商城等均采用二级（或多级）目录访问形式，这些网址均支持自定义设置（注意：系统下的各模块可以使用二级域名，各子模块使用的网址必须是同一个根域名），这里假设使用的根域名为 shopnc.net，系统标准域名配置参考如下：

说明	CONFIG 配置项	网址	指向目录
商城	shop_site_url	http://www.shopnc.net	shop
CMS	cms_site_url'	http://cms.shopnc.net	cms
圈子	circle_site_url	http://circle.shopnc.net	circle
微商城	microshop_site_url	http://microshop.shopnc.net	microshop
后台管理	admin_site_url	http://system.shopnc.net	admin
图片	upload_site_url	http://img.shopnc.net	data/upload
CSS,JS,IMG 等资源	resource_site_url	http://resource.shopnc.net	data/resource

以上二级域名也可自行命名，建议 www 指向商城作为主站，除以上提到的指向目录及子目录以外，其它目录强烈建议均不开放域名指向。

4. 附录

4.1 配置文件介绍

ShopNC B2B2C 商城包含多个独立模块，各独立模块均有自己的配置文件，配置文件整理如下：

说明	目录位置
全局配置文件[1]（需手动编辑）	data/config/config.ini.php
全局配置文件[2]（后台可自动编辑）	data/cache/setting.php
后台管理配置文件	admin/config/config.ini.php
CMS 配置文件	cms/config/config.ini.php
圈子配置文件	circle/config/config.ini.php
微商城配置文件	microshop/config/config.ini.php
IM 配置文件	chat/config/config.ini.php
全文搜索配置文件	data/api/xs/app/shopnc.ini

全局配置文件[1] 主要存放服务器级配置信息和不频繁改动的配置信息，如系统网址、数据库账号配置、redis 缓存配置等，如需更改，需要进入服务器手动编辑配置文件。

全局配置文件[2] 主要存放商城业务运营相关信息，如网站 LOGO，QQ 互联信息、上传附件信息等，该配置可以直接从管理后台相关操作界面更改。

商城、后台管理、圈子、CMS 和微商城各自目录下均有自己的配置文件，目前配置内容较少或无内容，预留二次扩展使用。

IM 和全文搜索具体配置可查看相关独立章节。

全局配置文件[1] 作为最主要的配置文件，配置参数与解释如下：

```
# 判断是否被合法调用
defined('InShopNC') or exit('Access Invalid!');

# 定义配置数组
$config = array();

# 商城网址
$config['shop_site_url'] = 'http://shop.shopnc.net';

# CMS 网址
```

```
$config['cms_site_url']          = 'http://cms.shopnc.net';

# 微商城网址
$config['microshop_site_url']    = 'http://microshop.shopnc.net';

# 圈子网址
$config['circle_site_url']       = 'http://circle.shopnc.net';

# 系统管理后台网址
$config['admin_site_url']        = 'http://system.shopnc.net';

# 图片网址
$config['upload_site_url']       = 'http://img.shopnc.net';

# 商城程序中 css, js, image 等资源网址
$config['resource_site_url']     = 'http://resource.shopnc.net';

# 系统版本号, 安装时自动生成
$config['version']               = '201401162490';

# 系统安装日期, 安装时自动生成
$config['setup_date']            = '2014-01-22 11:06:21';

# 是否开启 gzip 压缩
$config['gip']                   = 0;

# 数据库连接驱动, 支持 mysqli (默认), mysql
$config['dbdriver']              = 'mysqli';

# 数据库表前缀
$config['tablepre']              = 'shop_';

# 主数据库配置, 只允许配置一台(单台数据库时, 只配置主数据库即可)
$config['db'][1]['dbhost']       = '192.168.1.10';
$config['db'][1]['dbport']       = '3306';
$config['db'][1]['dbuser']       = 'root';
```

```
$config['db'][1]['dbpwd']      = 'root';
$config['db'][1]['dbname']     = 'shopnc';
$config['db'][1]['dbcharset']  = 'UTF-8';

# 从数据库配置，可以配置多台(多台时，系统每次随机使用)

# 如果没有从数据库，可以使用以下配置
$config['db']['slave']        = array();

# 如果使用从数据库，可以参考以下配置

# 第 1 台配置
$config['db']['slave'][1]['dbhost']   = '192.168.1.11';
$config['db']['slave'][1]['dbport']   = '3306';
$config['db']['slave'][1]['dbuser']   = 'root';
$config['db']['slave'][1]['dbpwd']    = 'root';
$config['db']['slave'][1]['dbname']   = 'shopnc';
$config['db']['slave'][1]['dbcharset'] = 'UTF-8';

# 第 2 台配置
$config['db']['slave'][2]['dbhost']   = '192.168.1.12';
$config['db']['slave'][2]['dbport']   = '3306';
$config['db']['slave'][2]['dbuser']   = 'root';
$config['db']['slave'][2]['dbpwd']    = 'root';
$config['db']['slave'][2]['dbname']   = 'shopnc';
$config['db']['slave'][2]['dbcharset'] = 'UTF-8';

# 第 N 台配置
.....

# 系统缓存默认时间，单(秒)，默认 1 小时，
$config['session_expire'] = 3600;

# 语言包，默认 zh_cn(简体)
$config['lang_type']      = 'zh_cn';
```



```
# cookie 前缀, 安装时自动生成
$config['cookie_pre']      = '08A6_';

# 模板目录
$config['tpl_name']        = 'default';

# 生成缩略图处理工具 可选为 gd(默认) 或 im, 分别代表 GD 库和 imagemagick
$config['thumb']['cut_type'] = 'gd';

# convert 可执行文件所在路径(位于 imagemagick 安装路径中), 只有使用 imagemagick 才需要配置该项, 使用 GD 留空即可
$config['thumb']['impath'] = '/usr/local/ImageMagick/bin';

# 缓存存储类型 支持类型为 file 文件缓存(默认)、redis、memcache、xcache、apc、eaccelerator
# 如果缓存类型设置为 memcache 或 redis, 还需要配置下面的 memcache 或 redis 相关参数才会生效
$config['cache']['type']   = 'file';

# memcache 设置, 可配置多台 memcache 服务器
$config['memcache']['prefix'] = 'nc_'; // memcached 存取时值时, 自动添加的前缀

# 第 1 台 memcached 配置
$config['memcache'][1]['port'] = 11211; // memcached 监听端口号
$config['memcache'][1]['host'] = '192.168.1.13'; // memcached 服务器 IP 地址
$config['memcache'][1]['pconnect'] = 0; //是否长连接 默认 0(否), 1 表示使用长连接

# 第 2 台 memcached 配置
$config['memcache'][2]['port'] = 11211;
$config['memcache'][2]['host'] = '192.168.1.14';
$config['memcache'][2]['pconnect'] = 0;

# 第 N 台 memcached 配置
.....

# redis 配置, 支持一主或一主一从配置
$config['redis']['prefix'] = 'nc_'; // redis 存取时值时, 自动添加的前缀

# redis 主服务器配置
```

```
$config['redis']['master']['port']      = 6379;
$config['redis']['master']['host']      = '192.168.1.15';
$config['redis']['master']['pconnect']  = 0;

# 不使用 redis 从服务器时, 配置如下
$config['redis']['slave']                = array();

# 使用 redis 从服务器时, 配置如下
$config['redis']['slave']['port']       = 6379;
$config['redis']['slave']['host']       = '192.168.1.16';
$config['redis']['slave']['pconnect']   = 0;

# 全文检索配置, 支持 true(开启) 或 false(关闭), 全文检索的详细配置, 请参考本帮助的全文检索章节

# 需编辑 data\api\xs\app\shopnc.ini, 将 server.index 和 server.search 值修改成自己的 IP 和端口
$config['fullindexer']['open']          = true;

# 全文检索配置文件名(默认为 shopnc, 不需要更改)
$config['fullindexer']['appname']       = 'shopnc';

# 是否开启调试模式, 支持 true(开启) 或 false(关闭)
# 开发环境建议开启调试, 开户后系统底部将显示详细的引入文件、运行时间、SQL 执行等信息, 如果发生异常, 系统直接输入
异常信息
# 生产环境建议关闭调试模式, 关闭后, 系统底部运行信息将不再显示, 如果发生异常, 系统会记录错误日志并保持系统继续运行

# 系统错误日志位于: data/log 目录
$config['debug']                        = false;

# 平台自营店店铺 ID
$config['default_store_id'] = '1';

# 是否开启伪静态, 支持 true(开启) 或 false(关闭)
$config['url_model'] = false;

# 二级域名后缀、SESSION、COOKIE 有效作用域
$config['subdomain_suffix'] = '';
```

后台管理配置文件内容及解析如下：

```
# 系统后台是否记录操作日志 true(开启) 或 false(关闭)
$config['sys_log'] = true;
```