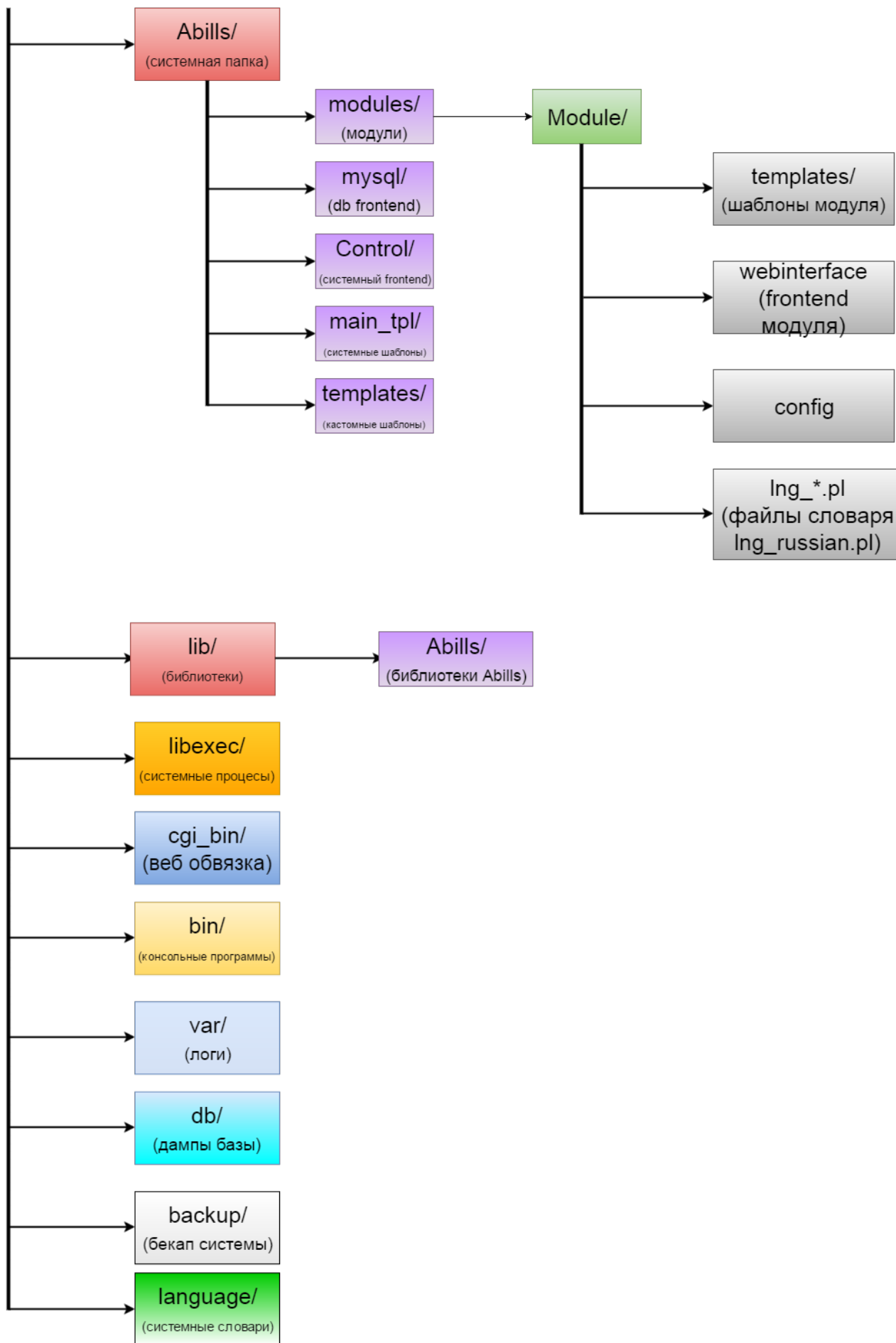


Модули системы

В данном разделе описано как создавать собственные модули для системы.

Структурная схема

ABILLS



Модуль для системы должны находиться в папке /usr/abills/Abills/modules. Название модуля лучше всего писать с заглавной буквы **New_module**. В каталоге модуля должно быть обязательно два файла webinterface и config.

- **webinterface** - Библиотека функций модуля
- **config** - Конфигурационный файл модуля

Включение модуля в конфигурационном файле: **libexec/config.pl**

```
@MODULES = (
  'Modulename'
);
```

config

Пример файла:

```
#Iner_ID : Main or Sub or if -1 user menu func list
%FUNCTIONS_LIST = ("01:0:$ _ICARDS:cards_main:" => 5,
                  "04:1:BRUTE FORCE:cards_bruteforce:" => 5,
                  "05:1:$ _ICARDS:cards_main:" => 5,
                  "06:1:$ _DILLERS $ _TARIF_PLANS:cards_dillers_tp:" => 5,
                  "06:5:$ _ADD:cards_add:" => 5,
                  "08:0:$ _ICARDS:cards_user_payment:UID" => 2,
                  "09:0:$ _ICARDS:cards_report:" => 4,
                  "11:8:$ _DILLERS:cards_report_dillers:" => 4,
                  "12:0:$ _DILLERS:cards_dillers_list:" => 1,
                  "13:0:$ _DILLERS:cards_diller:UID" => 15,
                  );

%USER_FUNCTION_LIST = ("01:0:$ _ICARDS:cards_user_payment:" => 0,
                      );

%RESELLER_FUNCTION_LIST = ("01:0:$ _ICARDS:cards_reseller_face:defaultindex" => 0,
                           "02:1:$ _REPORTS:cards_diller_stats:" => 1
                           );

%PERIODIC = (daily => [ 'cards_bruteforce_clean' ] );
```

В данном файле можно объявлять следующие переменные которые влияют на работу и отображение модуля в системе.

- **%FUNCTIONS_LIST** Список доступных функций в интерфейсе администратора. Каждая функция объявляется отдельной переменной хеш массива и имеет следующую структуру. Рассмотрим на примере:

```
%FUNCTIONS_LIST = ("01:0:$ _ICARDS:cards_main:UID" => 5);
```

В данном примере объявлена одна функция визуализации.

01	Номер функции в данном модуле
0	Номер родительской функции. Если 0 (ноль) функция является корневой. Если значение больше нуля то функция располагается в меню под родительской функцией
\$ _ICARDS	Название функции в меню панели управления
cards_main	Название функции которая выполняется при нажатии соответственного меню

UID	Дополнительные параметры. Они могут быть и не объявлены. Если объявляются тогда меню показывается только если данные доп параметры присутствуют HTTP запросе. Например ели объявлен параметр UID то данная функция будет отображаться только если в запросе он присутствует. Пример запроса: http://your.host:9443/index.cgi?UID=15
5	Номер корневого каталога в котором разместить меню. Номера корневых меню: 1 - Клиенты 2 - Оплаты 3 - Снятие денег 4 - Отчёт 5 - Настройка 6 - Мониторинг 7 - Поиск 8 - Обслуживание 9 - Профиль

- **%USER_FUNCTION_LIST** - Список функций личного кабинета пользователя. Функции формируются за тем же правилом что и функции интерфейса администратора.
- **%RESELLER_FUNCTION_LIST** - Список функций интерфейса реселлера. Интерфейс реселлера появляется только при включении модуля Multidoms
- **%PERIODIC** - Список функций которые используются в периодических процессах системы. Они вызываются из программы `abills/libexec/periodic`.

В системе доступно несколько видов функций:

daily	Дневные переодические функции
monthly	Месячные переодические функции
reports	Функции составления дневные отчётов

пример объявления:

```
%PERIODIC = (daily => [ 'dv_daily_fees', 'dv_users_warning_messages' ],
              monthly => [ 'dv_monthly_fees', 'dv_periodic_logrotate' ],
              report => [ 'dv_report' ] );
```

Проверить

дневной периодик

```
/usr/abills/libexec/periodic daily MODULES=[module name]
```

webinterface

В данном файле должны располагаться функции модуля

Языковые файлы

По умолчанию система использует файлы **language/english.pl** и региональные `ukraine.pl`, `russian.pl` и т. д. Если нужно для модуля создать собственные словари они должны размещаться в папке модуля с названиями `lng_xxx.pl`.

Например:

```
lng_english.pl
lng_ukraine.pl
lng_russian.pl
```

Если модуль имеет свои собственные языковые файлы система автоматически их подгружает

Создание быстрых отчётов для первой страницы

На первой странице системы можно разместить несколько быстрых отчётов. Какие отчёты размещать определяет администратор в меню профайл. Ниже будет описано как создать быстрый отчёт в созданном вами модуле.

Для того чтобы сообщить системе что модуль (в нашем примере название модуля `New_module`) имеет быстрые отчёты нужно создать функцию

под названием **`New_module_start_page`** Данная функция должна возвращать ссылку на ассоциированный массив значений в котором ключ это название функции выводящей быстрое меню а значение её описание.

Пример:

```
sub new_module_start_page {
    my ($attr) = @_;

    my %START_PAGE_F = ('new_module_sp_report' => 1 );

    return \%START_PAGE_F;
}
```

создаём сам отчёт

```
sub new_module_sp_report {

    print "Hello, World";

}
```

Работа с базой данных

Для разделения работы с базой и элементов визуализации все функции работы с базой помещаются в каталог `/usr/abills/Abills/[название базы данных]`
Для СУБД `mysql` это `/usr/abills/Abills/mysql`.

Функция добавления

`query_add($table, $values, $attr)` - Добавление данных в базу.

Аргументы:

<code>\$values</code>	хеш ключей и значений, например <code>[COL_NAME] => [value]</code>
<code>\$table</code>	название таблицы
<code>\$attr</code>	дополнительные атрибуты

Дополнительные атрибуты:

<code>REPLACE</code>	Меняет команду <code>INSERT</code> на <code>REPLACE</code>
-----------------------------	--

Пример:

```
$self->query_add( 'table_name', {COL_NAME_1 => VALUE_1,
                                COL_NAME_2 => VALUE_2,
                                COL_NAME_3 => VALUE_3}, {REPLACE => 1} );
```

Функция удаления

query_del(\$table, \$values, \$extended_params, \$attr) - удаление данных из базы.

Аргументы:

\$table	название таблицы
\$values	параметры удаления
\$extended_params	расширенные параметры удаления, хеш ключей и значений, например [COL_NAME] => [value]
\$attr	дополнительные параметры

Дополнительные параметры:

CLEAR_TABLE	очищается таблицу целиком
--------------------	---------------------------

Пример:

```
$self->query_del('table_name', {COL_NAME => VALUE});
$self->query_del('table_name', {}, {CLEAR_TABLE => 1});
```

Функция изменения

changes2(\$attr) - вносит изменения в базу

\$attr	параметры, хеш ключей и значений
---------------	----------------------------------

Параметры:

CHANGE_PARAM	параметр изменения, изначально ID (обязателен)
SECOND_PARAM	дополнительный параметр изменения
TABLE	название таблицы (обязателен)
DATA	новые данные (хеш ключей и значений)
EXT_CHANGE_INFO	
FIELDS	
OLD_INFO	
ACTION_ID	

Пример:

```
$self->changes2(
{
    CHANGE_PARAM => 'ID',
    TABLE       => 'TABLE_NAME',
    DATA        => {COL_NAME_1=>VALUE_1, COL_NAME_2 => VALUE_2}
}
);
```

Другие команды в базу

query2(\$query, \$type, \$attr) - выполнение своих команд в базу

Параметры:

\$query	запрос
\$type	тип запроса, undef - with fetch result like SELECT, do - do query without fetch (INSERT, UPDATE, DELETE)
\$attr	дополнительные параметры

Дополнительные параметры:

COLS_NAME	Возвращает ссылку на хеш. Имя колонки как ключ хеша
COLS_UPPER	Сделать ключи большими буквами
INFO	Return fields as objects parameters \$self→{LOGIN}
LIST2HASH	Возвращает хеш, вместо листа {KEY⇒VAL}
MULTI_QUERY	Мультизапросы (только для INSERT, UPDATE)
Bind	Массив или бинд значений [10, 12, 33]
DB_REF	Объект БД. Используется когда работа идет с несколькими ДБ серверами
test	Исполняется без запроса. Если исползуется \$self→{debug} показывается запрос.

Пример:

```

$self->query2(
    "SELECT * FROM table_name;",
    undef,
    {}
);

```

HTML.pm

Модуль для вывода HTML контента

Шаблоны

Для визуализации используется система шаблонов

отображение глобального шаблона системы с названием **form_chg_tp**. Глобальные шаблоны системы размещаются в каталоге **Abills/main_tpls/**.

```

print $html->tpl_show(templates('form_chg_tp'), $attr, { ... });

```

Функция **templates()** ищет шаблон в каталоге кастомных шаблонов а также смотрит есть ли языковой шаблон данного типа.

параметр **\$attr** передаёт обработчику шаблонов ссылку на массив значений маркеров переменных

отображение шаблона для модуля (шаблоны модулей сохраняются в каталоге **Abills/module/[module_name]/templates**)


```
print $html->tpl_show(_include('dv_users_search', 'Dv'), $attr);
```

Функция **_include()** загружает шаблон **dv_users_search** для модуля **DV**

Дополнительные возможности

Проверка значения input на соответствие шаблону

Если указать атрибут **data-check-for-pattern=«RegExp»**, где **RegExp** - JavaScript шаблон, то при вводе значение будет проверяться на соответствие паттерну.

Если значение не соответствует паттерну, родительскому элементу добавляется класс **has-error**.

Текст **label** и рамка **input** меняют цвет на красный.

Если значение соответствует шаблону, класс **has-error** снимается.

Шаблоны JavaScript тут: <http://javascript.ru/RegExp>

Для полей ввода IP-адресов, достаточно добавить класс **ip-input**. Автоматически добавится паттерн для проверки корректности IPv4 или IPv6 адреса.

Ввод значения в один input при доступных нескольких

Добавить атрибут **data-input-disables=%ID%** в **<input>** (**<select>**, **<input type='checkbox'>**)

При вводе значения, удаляет значение в **<input id='%ID%'>** и добавляет свойство **disabled**.

Если указать **id** тега **<fieldset>**, отключит все **<input>** внутри тега.

Можно указать несколько **id** через запятую (**без пробелов**).

Обратное поведение (включить поле ввода по значению), можно используя атрибут **data-input-enables=%ID%**

Всплывающая подсказка при наведении мыши

Добавьте элементу атрибут **data-tooltip** для отображения всплывающей подсказки.

Значением атрибута могут быть любой HTML код.

Атрибут **data-tooltip-position** управляет позицией **tooltip**'а, возможные значения **left**, **right**, **top**, **bottom**.

Решение проблемы с checkbox

По умолчанию, браузер не отправляет неотмеченный **checkbox**, иногда это может быть неудобно.

Если **checkbox** используется в форме, где нужно отсылать неотмеченное состояние, укажите атрибут **data-return='1'**

AJAX загрузка файла

Шаблон модального окна

Select с автотправкой

Автоматически отправляет форму после изменения. Доступны 2 варианта

- Отправка GET по указанному линку. &COLORS=%значение селекта%, добавится автоматически

```
my $colors_css = $html->form_select( 'COLORS',
{
  SELECTED => $admin->{SETTINGS}->{COLORS} || q{},
  SEL_ARRAY => \@names,
  NO_ID     => 1,
  SEL_OPTIONS => { '' => $lang{DEFAULT} },
  EX_PARAMS => "data-auto-submit='index=$index&AWEB_OPTIONS=1'"
}
);
```

- Отправка формы, в которой находится <select>

```
my $custom_select = $html->form_select( 'CUSTOM',
{
  SELECTED => $FORM{CUSTOM} || q{},
  SEL_ARRAY => \@options,
  NO_ID     => 1,
  SEL_OPTIONS => { '' => $lang{NO} },
  EX_PARAMS => "data-auto-submit='form'"
}
);
```

Условное скрытие/показ элемента

Доступно через атрибут **data-visible**.

Любое значение интерпретируемое в JS как **false** добавит **display : none**

Любое значение интерпретируемое в JS как **true** добавит **display : block**

Отображение даты в человекочитаемом формате

Отобразить время от/до наступления даты (3 часа назад, 5 дней назад, Через 3 дня)

```
<span class='moment-insert' data-value='YYYY-MM-DD hh:mm:ss'></span>
```

Отобразить временной интервал (3 часа, 5 дней)

```
<span class='moment-range' data-value='ss'></span>
```

Подтверждение действий в кабинете пользователя

Есть возможность добавить в кабинете пользователя подтверждение действий паролем. Для этого необходимо добавить форме класс **pswd-confirm**. При попытке сабмита формы с этим классом, откроется модальное окно для ввода пароля. Введенный пароль добавится к данным формы. (работает при включенной опции `$conf{user_confirm_changes}=1;`)

result_former

Функция формирования результирующей таблицы используя SQL, XML JSON или HASH данные

```

my ($table, $list) = result_former({
  INPUT_DATA      => $Dv,
  FUNCTION        => 'list',
  BASE_FIELDS     => ,
  DEFAULT_FIELDS => 'LOGIN,FIO,DEPOSIT,CREDIT,TP_NAME,DV_STATUS',
  FUNCTION_FIELDS => 'form_payments,dv_stats, del, change',
  EXT_TITLES      => {
    'ip'           => 'IP',
    'netmask'      => 'NETMASK',
    'speed'        => $_SPEED,
    'port'         => $_PORT,
    'cid'          => 'CID',
    'filter_id'    => 'Filter ID',
    'tp_name'      => "$_TARIF_PLAN",
    'dv_status'    => "Internet $_STATUS",
    'dv_status_date' => "$_STATUS $_DATE",
    'online'       => 'Online',
    'dv_expire'    => "Internet $_EXPIRE",
    'dv_login'     => "$_SERVICE $_LOGIN",
    'dv_password' => "$_SERVICE $_PASSWD"
  },
  TABLE          => {
    width          => '100%',
    caption        => "Internet - $_USERS",
    qs             => $_pages_qs,
    ID             => 'INTERNET_USERS_LIST',
    header         => $status_bar,
    EXPORT         => 1,
    MENU           => "$_ADD:index=" . get_function_index('dv_wizard_user') . ':add' .
";$_SEARCH:index=$index&search_form=1:search",
  },
  MAKE_ROWS       => 1,
  SEARCH_FORMER   => 1,
  MODULE          => 'Dv',
  TOTAL           => 1
});

```

Карта покрытия

Если у системе подключён модуль Maps для объектов имеющих возможность гео привязки можно рисовать карту покрытия

Для включения опции карты параметр

```
MAP => 1,
```

Также можно указать информацию для отображения на сводных таблицах

```
MAP_FIELDS => 'ID,CLIENT_ID,SUBJECT'
```

И фильтр обработки параметров полей сводной таблицы

```
MAP_FILTERS => { id => 'search_link:msgs_admin:UID,chg={ID}',
}
```

При отображении элементов на карте система устанавливает цвета точек гео локации

Белый	нет привязанных объектов к точке
Зелёный	не больше 3 привязанных элементов к точке геолокации
Синий	Больше 3 меньше 10
Жёлтый	Больше 10 меньше 20
Красный	Больше 20
Огонь	Критическое состояние точки

Графики

```

$html->make_charts(
    {
        PERIOD      => $graph_type,    #Период графика,
        DATA       => \%DATA_HASH,
        AVG         => \%AVG,
        TRANSITION  => 1,
        TYPE        => {column_name => 'type' },
        X_LINE      => [ $val ],
        X_TEXT      => [ $val ],
        DATA       => { cols_name => [ $val ] };
        OUTPUT2RETURN => 1,
    }
);
    
```

PERIOD

Период и тип графиков

- week** - недельный с пн по пт
- year** - Год
- month** - месячный от 1 до 31 числа
- hour** - почасовой график 0 до 24 часов
- pie** - круг

AVG

Усреднённые значения

TRANSITION

Анимация визуализации

TYPES

⇒ {column_name ⇒ 'type' },

тип отображения значений поля на графике

column_name - название колонки

type - тип:

LINE	Линия
COLUMN	колонка на колонках
AREA	Заполнение
SCATTER	Отображение точками
PIE	Отображение параметров поля круговым графиком

X_LINE

⇒ [\$val],

Текст X оси

X_TEXT

⇒ [\$val],

Текст X оси

DATA

⇒ { cols_name ⇒ [\$val] };

массив значений

cols_name - название колонки

\$val - список значений для отображения

Если задана \$attr→{PERIOD}, то в зависимости от значения, формируются категории (X_TEXT) и корректируется длина массива (отрезается 0-ой индекс).

'week_stats'	['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun']	7
'month_stats'	[1,2,3,4,5,6,7,8,9,...,31]	31
'year_stats'	['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec']	12
'day_stats'	['00:00:00', '01:00:00', '02:00:00', ..., '24:00:00']	24

OUTPUT2RETURN

отдать значение в STDO

Полезные функции WEB интерфейса

- **get_function_index('form_info_fields')**

Получить index функции.

Maps

[Документация разработчика Maps](#)

JavaScript

[Документация по JS библиотекам](#)

Статические значения qindex

Поскольку, все значения индексов формируются во время выполнения, есть необходимость резервировать index'ы некоторых функций.

- 100000 - QRCode former
- 100001 - msgs_register_push_client
- 100002 - form_events (Клиентский интерфейс)

Подсказки к шаблонам

Начиная с версии 0.77.66 можно заполнять дополнительные подсказки для полей шаблона.
При использовании новой схемы отображения шаблона

```
$html->tpl_show('', {  
  VAR1 => 'VAL1',  
},  
{  
  TPL    => 'module_entity',  
  MODULE => 'Module',  
  HELP   => 1  
});
```

Если задан ключ **HELP** ⇒ **1**, к шаблону добавится контент файла `module_entity_help.tpl`, который будет обработан на странице.

Формат файла шаблона подсказок: ID:комментарий.

Пример:

```
MODULE: Имя модуля  
EXTRA: Ссылка для получения доп. информации  
COMMENTS_id: Комментарии (текст уведомления)
```

From:
<http://abills.net.ua/wiki/> - **Advanced Billing Solution**

Permanent link:
<http://abills.net.ua/wiki/doku.php/abills:docs:development:modules:ru>

Last update: **2019/01/26 09:22**

