

Бillion PBX

Программная IP-АТС

Версия 2.1

Система ИНТЕРАКТИВНЫХ ГОЛОСОВЫХ КОМАНД (IVR)

Содержание

1. Введение	3
Должен ли администратор Billion PBX уметь писать IVR-скрипты?.....	3
2. Язык написания IVR-скриптов	3
2.1. Общие сведения	3
2.2. Расположение файлов с IVR-скриптами.....	6
2.3. Использование IVR-скриптов	6
3. Справочник системы IVR	6
3.1. Служебные переменные	6
3.2. Блок IVR	8
3.3. Блок PlayMessages.....	8
3.4. Блок RecordMessage	11
3.5. Блок DeleteMessages	12
3.6. Блок TransferCall	13
3.7. Блок EndCall	14
3.8. Блок ParseFolder	15
3.9. Блок Assign	15
3.10. Блок SetLanguage.....	16
3.11. Блок StoreVariable	16
3.12. Блок RetrieveVariable	18
3.13. Блок ChangeNumber.....	18
3.14. Блок SumStrings	18
3.15. Блок CompareStrings	19
3.16. Блок CompareNumbers.....	20
3.17. Блок Input	21
3.18. Блок Choice	22
3.19. Блок Billing.....	23
3.20. Блок OS_Shell	25
3.21. Блок VoucherIvr.....	26
3.22. Блок DateSwitchIvr	27
3.23. Блок Switch.....	29
4. IVR-скрипты, входящие в дистрибутив	30

1. Введение

В состав Billion PBX входит развитый процессор скриптов для организации интерактивного голосового меню. IVR-скрипты описывают сценарии работы таких функций как голосовая почта, обратный вызов и др. Администраторы Billion PBX могут создавать собственные IVR-скрипты, реализующие требуемую логику взаимодействия с абонентом.

Данный документ является учебным и справочным пособием по системе IVR. В нем рассматриваются принципы работы системы IVR, приводится подробное описание всех элементов, используемых в IVR-скриптах, а также даются примеры IVR-скриптов, входящих в дистрибутив Billion PBX.

Документ ориентирован на читателя, имеющего общее представление о программировании на процедурных языках и знакомого с XML. Кроме того, предполагается, что читатель понимает общие принципы работы Billion PBX, описанные в документе «Знакомство с Billion PBX».

Должен ли администратор Billion PBX уметь писать IVR-скрипты?

Не обязательно. В состав Billion PBX входит широкий набор готовых скриптов, которых может быть достаточно для реализации всех требуемых схем взаимодействия с абонентом.

Однако при умении пользоваться языком написания IVR-скриптов, перед администратором открываются широчайшие возможности по созданию голосовых меню, максимально полно отражающих требования к используемой телефонной системе.

2. Язык написания IVR-скриптов

2.1. Общие сведения

IVR-скрипты являются документами в формате XML. Первая строка каждого скрипта обязательно должна выглядеть следующим образом:

```
<?xml version="1.0"?>
```

Для работы с IVR-скриптами можно использовать текстовый редактор notepad.exe или специализированные средства, например, Microsoft Visual Studio.

IVR-скрипты могут быть как самыми простыми, например, которые всего лишь проигрывают медиа-файл, так и реализующими достаточно развитую логику, включая получение от пользователя управляющих команд с помощью DTMF-сигналов и запись голосовых сообщений.

IVR-скрипт состоит из блоков. Каждый блок начинается с открывающего XML-тэга и заканчивается закрывающим. Вся логика IVR-скрипта помещается в главный блок IVR, внутри которого содержатся остальные блоки:

```
<?xml version="1.0"?>
<IVR>
... логика IVR-скрипта
</IVR>
```

В приведенном ниже примере в главном блоке IVR-скрипта содержится один блок – PlayMessages.

```
<?xml version="1.0"?>
<IVR>
<PlayMessages>
<File value="BusyMessage" repeat="10"/>
</PlayMessages>
</IVR>
```

Блоки обрабатываются сверху вниз, если не указаны команды перехода, о которых будет рассказано позже. В следующем примере будут последовательно выполнены два блока PlayMessages.

```
<?xml version="1.0"?>
<IVR>
<PlayMessages>
<File value="WelcomeMessage" />
</PlayMessages>
<PlayMessages>
<File value="BusyMessage" repeat="10"/>
</PlayMessages>
</IVR>
```

Таким образом, блоки в IVR-скрипте являются аналогами команд в процедурных языках программирования.

Открывающий блок может также содержать параметры, например,

```
<IVR first="InputNumber" listenMode="Half">
```

Каждый блок, кроме блока IVR позволяет указывать следующие параметры:

Параметр	Описание
Name	Произвольное имя блока, которое может быть использовано, например, для перехода из другого блока.
Next	Имя блока для перехода после выполнения данного блока. Если задан данный параметр, то по окончании выполнения текущего блока, будет выполнен блок, параметр Name которого равен указанному здесь значению, несмотря на то, что этот блок может и не быть следующим по порядку сверху вниз.

Следует обратить внимание, что для некоторых блоков указание описанных выше параметров может не иметь смысла, например, для блока EndCall. Кроме того, некоторые блоки осуществляют переход не на один, а на несколько блоков, в зависимости от выполненных условий, например, блоки Choice, Switch, DateSwitch, CompareStrings, CompareNumbers, Billing и TransferCall.

В следующей таблице приводится список блоков, используемых в IVR-системе Billion PBX, и их краткое описание. Более подробно описание приведено в разделе [Справочник системы IVR](#).

Блок	Описание
IVR	Главный блок в IVR-скрипте. Включает в себя все остальные блоки.
Assign	Присваивает значение переменной.
Billing	Выполняет запрос на биллинг.
ChangeNumber	Изменяет значение числовой переменной.
Choice	Передаёт управление в зависимости от введённого пользователем DTMF.
CompareStrings	Сравнивает строки или строковые переменные.
CompareNumbers	Сравнивает числа или числовые переменные.
DateSwitchIvr	Управляет логикой скрипта в зависимости от параметров даты и времени.
DeleteMessages	Удаляет сообщения.
EndCall	Завершает вызов.
Input	Принимает dtmf-сообщение от пользователя.
OS_Shell	Выполняет команду операционной системы или командный файл.
PlayMessages	Проигрывает сообщения пользователю.
RecordMessage	Записывает сообщения пользователя.
ParseFolder	Ищет сообщения в директории.
RetrieveVariable	Извлекает значение переменной из постоянного хранилища.
SetLanguage	Устанавливает язык IVR.
StoreVariable	Сохраняет значение переменной в постоянном хранилище.
Switch	Управляет логикой скрипта в зависимости от содержания переменных.
SumStrings	Суммирует строки или строковые переменные.
TransferCall	Иницирует переадресацию вызова оборудованием пользователя на другой номер и IP-адрес.
VoucherIvr	Пополняет баланс абонента с использованием ваучера.

Внутри каждого блока может быть задан набор атрибутов, которые определяют поведение этого блока. Как правило, атрибут – это одиночный XML-тэг, содержащий набор параметров. В следующем примере в блоке `PlayMessages` задается атрибут `File`, который, в частности, и определяет, какой именно звуковой файл будет проигрываться в данном блоке.

```
<PlayMessages>  
<File value="WelcomeMessage" />  
</PlayMessages>
```

Состав атрибутов, используемых в каждом блоке, приводится в разделе [Справочник системы IVR](#).

2.2. Расположение файлов с IVR-скриптами

Файлы IVR скриптов находятся в директории ivr, файлы звуковых сообщений – в директории ivr/ru/wav.

2.3. Использование IVR-скриптов

Для того чтобы Billion PBX использовал IVR-скрипты, необходимо указать их в соответствующих полях Панели управления.

Существует два способа направить вызов на IVR-скрипт:

- Для Внешнего устройства в параметрах оригинации есть поле «Набранный номер для использования IVR». Если набранный номер совпадает с введённым (в том числе, пустой), то управление будет передано IVR скрипту из файла, указанного в параметре «Имя файла скрипта IVR» (по умолчанию, CardPlatform).
- Создать внешнее устройство с протоколом «IVR», указав имя IVR-скрипта в поле «IP адрес или имя». Можно не указывать имя IVR скрипта, а сразу написать имена файлов для воспроизведения, разделённые пробелами. Тогда BPBX будет использовать Msg.xml скрипт.

При вводе наименования IVR-скрипта в полях Панели управления Billion PBX могут быть также заданы входящие параметры. Набор принимаемых параметров зависит от скрипта. В следующем примере вызывается IVR-скрипт EndCall, которому передаются два параметра, которые используются внутри скрипта:

```
EndCall.xml;q931=34;sip=600
```

3. Справочник системы IVR

3.1. Служебные переменные

В скрипте используются переменные. Переменная имеет имя, по которому к ней можно обратиться, и значение. Существует два типа переменных – временные, доступные только в текущем исполнении данного скрипта (в дальнейшем называемая просто переменная), и постоянные, доступные во всех исполнениях данного скрипта. Постоянные переменные хранятся в файле с тем же именем, что имя IVR-скрипта и расширением .store.

В следующей таблице приводятся переменные, которые инициализируются в начале выполнения скрипта.

Переменная	Описание
billing-user billing-password	Если в эти переменные в процессе исполнения IVR что-нибудь занесено, то это заместит логин и пароль для биллинга, текущие для данного внешнего устройства. Это имеет смысл только для IVR, который вызывается до

	биллинга. Данные переменные также используются функцией TransferCall и могут быть изменены перед переводом звонка.
billing-type	Может быть пустым или принимать следующие значения: RADIUS, Internal.
call-end-cause-bpbx	BPBX-код завершения вызова.
call-end-cause-sip	SIP-код завершения вызова.
call-end-cause-q931	Q931-код завершения вызова.
call-id	Уникальный идентификатор вызова.
called-number	Набранный телефонный номер или имя пользователя, которому адресован вызов.
caller-number	Имя или телефонный номер инициатора.
caller-address	IP-адрес инициатора.
caller-id	ID инициатора
called-number-prefix	Префикс в набранном номере.
called-number-extension	Набранный номер без префикса.
customer	Клиент внутреннего биллинга.
date	Текущая дата.
date-minute	Минуты в текущем времени.
date-hour	Часы в текущем времени.
date-day	День в текущей дате.
date-day-of-week	Текущий день недели: воскресенье = 0, понедельник = 1 и т.д.
date-is-weekend	0 – если текущий день – суббота или воскресенье, в противном случае 1.
date-month	Текущий месяц: январь = 1, февраль = 2 и т.д.
date-year	Текущий год.
is-message	1 - если sms, 0 - если голосовой вызов.
ivr-name	Имя файла ivr-скрипта.
Language	Язык (Russian, English).

message-text	Текст sms.
--------------	------------

3.2.Блок IVR

Блок IVR является главным блоком IVR-скрипта. Он включает в себя все остальные блоки и в этом смысле является особенным. В каждом файле может быть только один блок IVR.

Блок IVR позволяет задать следующие параметры:

Параметр	Описание
first	Определяет имя блока, который будет выполняться при запуске IVR-скрипта. Если данный параметр не указан, первым будет выполнен самый верхний блок.
listenMode	<p>Определяет режим приема медиа-сигналов от абонента в процессе выполнения IVR-скрипта. Возможные значения:</p> <ul style="list-style-type: none"> • None – прием сигналов от абонента не производится, • Half – прием сигналов от абонента производится, но медиа-поток на раскодируется – возможен приём DTMF только в формате RFC2833, • Full – производится раскодирование медиа-потока от абонента – возможен приём DTMF методом in-band. При этом невозможна передача уже закодированных и сохраненных в wav-файлах сообщений, например, в формате кодека G.729. • Absent – медиа-поток не открывается. Используется в IVR-скрипте с единственным блоком EndCall. <p>Если данный параметр не указан, Billion PBX будет анализировать содержимое IVR-скрипта. В случае наличия в скрипте блока RecordMessage, будет включен режим Full. В случае наличия в скрипте блоков Input или Choice, будет включен режим Half.</p>

3.3.Блок PlayMessages

Блок PlayMessages проигрывает сообщения пользователю.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
File	<p>Задаёт файл для проигрывания. В блоке может быть несколько экземпляров данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • value – имя файла сообщения без расширения, относительно директории /ivr/Russian/wav. Может содержать несколько имен,

	<p>разделенных пробелами. В этом случае файлы будут проигрываться последовательно.</p> <ul style="list-style-type: none"> • vary – имя переменной, в которой хранится имя файла сообщения без расширения, относительно директории /ivr/Russian/wav. Может содержать несколько имен, разделенных пробелами. В этом случае файлы будут проигрываться последовательно. • repeat – количество повторений. По умолчанию 1. Если -1, файл будет проигрываться бесконечно. Если введено не число, предполагается, что это – имя переменной.
Folder	<p>Атрибут Folder подобен атрибуту File, но определяет директорию, из которой будут проигрываться все сообщения. Дополнительно он имеет параметр includeSubFolders, который определяет, проигрывать ли сообщения, содержащиеся в поддиректориях. По умолчанию этот параметр равен <i>False</i>.</p>
Number	<p>Атрибут Number позволяет проговорить число. В блоке может быть несколько экземпляров данного атрибута.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • value – число, • vary – имя переменной, хранящей число, • descr – описывает переменную с описанием – т.е., что именно означает эта цифра. • descrlsKey – описывает, берётся ли значение из переменной или из параметра. <p>При выполнении этого блока по умолчанию понадобится 2 звуковых файла описания – для единственного и множественного числа, причём имя файла с единственным числом равно имени файла с множественным числом минус последний символ. Правила склонения цифр и описания по правилам русского языка можно посмотреть в файле <code>ivr/ru/number.exceptions</code>. Он содержит описание для времени (секунда, минута, час и день). Аналогичные описания для своих имен можно поместить или в этот файл или задать блоком Assign.</p>
TelNumber	<p>Позволяет проговорить телефонный номер. В блоке может быть несколько экземпляров данного атрибута.</p> <p>По умолчанию он проигрывается в виде последовательности цифр – например, 5428518 будет проигран как 5 4 2 8 5 1 8. В файле <code>ivr/ru/telnumber.patterns</code> можно задать формат проигрывания номера.</p>

	<p>Например, строка 7=3:2:2 относится ко всем 7-значным номерам и номер 5428518 будет проигран как 542 85 18. Сумма модулей цифр после знака равно должна быть равно числу перед этим знаком. Если цифра после знака равно меньше нуля, то эта группа цифр проигрывается не будет. Например, строка 10=-1:3:3:2:2 означает, что номер 78125428518 будет проигран как 812 542 85 18.</p>
TimeInterval	<p>Позволяет проговорить временной интервал, заданный в секундах. В блоке может быть несколько экземпляров данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • value – число, • vary – имя переменной, хранящей число.
Money	<p>Позволяет проговорить денежную сумму с указанием десятичных единиц (копеек) или без их указания. В блоке может быть несколько экземпляров данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • value – число, • vary – имя переменной, хранящей число. • Cents – допустимые значения: <i>True</i> или <i>False</i> – включает и выключает режим проговаривания остатков после десятичной точки (копеек).
Silence	<p>Позволяет установить период бездействия. В блоке может быть несколько экземпляров данного атрибута. Данный атрибут имеет два параметра:</p> <ul style="list-style-type: none"> • secs – время в секундах, • msecs – время в миллисекундах. <p><Silence secs="5"/></p>
Date	<p>Позволяет проговорить дату. В блоке может быть несколько экземпляров данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • value – дата, • vary – имя переменной, хранящей дату. • year – допустимые значения: <i>True</i> или <i>False</i> – если значение равно <i>True</i>, то проговаривается год. По умолчанию – <i>True</i>. • weekday – допустимые значения: <i>True</i> или <i>False</i> – если значение равно <i>True</i>, то проговаривается день недели. По умолчанию – <i>True</i>.
Time	<p>Позволяет проговорить время. В блоке может быть несколько экземпляров</p>

	<p>данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • value – время, • vary – имя переменной, хранящей время. • seconds – допустимые значения: <i>True</i> или <i>False</i> – если значение равно <i>True</i>, то проговариваются секунды. По умолчанию – <i>False</i>.
--	--

В приведенном ниже примере 2 раза проигрывается файл `ivr/ru/wav/SubFolder/FileName.wav`, проигрываются все сообщения из директории `ivr/ru/wav` (предполагается, что значение переменной **FolderName** было присвоено ранее), проигрывается количество дней из переменной **Number**, деньги из переменной **Money**, временной интервал в 1234 секунды и АОН.

```
<PlayMessages name="PlayMessage" next="NextBlock">
<File value="SubFolder/FileName" repeat="2"/>
<Folder vary="FolderName" includeSubFolders="False" repeat="1"/>
<Number vary="Number" descr="days" descrIsKey="True"/>
<Money vary="money"/>
<TimeInterval value="1234"/>
<TelNumber value="caller-number"/>
<Date vary="date" year="True" weekday="True"/>
<Time vary="date" seconds="True"/>
</PlayMessages>
```

3.4. Блок RecordMessage

Блок RecordMessage сохраняет сообщение пользователя в файле. Если файл уже существует, то сообщение дописывается в конец файла.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
File	<p>Задаёт файл для записи. Параметры:</p> <ul style="list-style-type: none"> • value – имя файла для записи без расширения, • vary – имя переменной, хранящей имя файла для записи без расширения.
Folder	Атрибут Folder подобен атрибуту File , но определяет директорию, в которую будут производиться запись. В этом случае имя файла будет присвоено автоматически.
Timeout	Время в секундах, по истечении которого запись прекращается. Задаётся параметром value .
Terminator	Набор чисел или знаков <code>#/*</code> , при введении которого посредством dtmf производится переход к следующему блоку. Задаётся параметром value .

Email	<p>Позволяет отправить записанное голосовое сообщение в виде wav-файла на указанный адрес электронной почты. Параметры:</p> <ul style="list-style-type: none"> • value – адрес электронной почты, • vary – имя переменной, хранящей адрес электронной почты.
Subject	<p>Определяет тему электронного сообщения при наличии атрибута Email. Параметры:</p> <ul style="list-style-type: none"> • value – тема электронного сообщения, • vary – имя переменной, хранящей тему электронного сообщения.
Body	<p>Определяет текст электронного сообщения при наличии атрибута Email. Параметры:</p> <ul style="list-style-type: none"> • value – текст электронного сообщения, • vary – имя переменной, хранящей текст электронного сообщения.
DeleteAfterSend	<p>Определяет, надо ли удалять файл с голосовым сообщением после отправки его по электронной почте при наличии атрибута Email.</p> <ul style="list-style-type: none"> • value – <i>True/False</i>. По умолчанию – <i>False</i>, • vary – имя переменной, хранящей <i>True/False</i>.

В приведенном ниже примере сообщение пользователя сохраняется в файле, имя которого содержится в переменной **FileName**. Ввод прекращается через 10 секунд или если пользователь введёт посредством dtmf “#”.

```
<RecordMessage name="RecordMessage" next="EndCall">
<File vary="FileName"/>
<Timeout value="10"/>
<Terminator value="#"/>
<Email vary="email"/>
<Subject vary="subject"/>
<Body value="Voice message attached to this email"/>
<DeleteAfterSend value="True"/>
</RecordMessage>
```

3.5.Блок DeleteMessages

Блок DeleteMessages удаляет звуковые сообщения. Его атрибуты аналогичны атрибутам Блока PlayMessages за исключением параметра **repeat**, который здесь не применяется.

В приведенном ниже примере удаляется файл ivr/Russian/wav/MessageName.wav – и все звуковые файлы из директории ivr/Russian/wav/FolderName и всех нижележащих директорий. Сами директории не удаляются.

```
<DeleteMessages name="DeleteMessage" next="NextBlock">
<File value="../MessageName"/>
<Folder value="FolderName" includeSubFolders="True"/>
</DeleteMessages>
```

3.6. Блок TransferCall

Блок TransferCall инициирует переадресацию вызова на заданное направление. Данный блок имеет набор дополнительных параметров, приведенных в следующей таблице:

Параметр	Описание
onFailed	Задаёт имя блока, которому будет передано управление в случае неудачного перевода вызова.

Атрибуты **File**, **Folder**, **Number**, **TelNumber**, **TimeInterval**, **Money**, **Silence**, **Date** и **Time** аналогичны соответствующим атрибутам в блоке [PlayMessages](#). В следующей таблице приводится набор дополнительных атрибутов, которые могут быть заданы в рамках данного блока.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
User	<p>Определяет номер для трансфера. Параметры:</p> <ul style="list-style-type: none"> value – номер телефона, на который будет переведен вызов, vary – имя переменной, хранящей номер телефона, на который будет переведен вызов.
FailedReason	<p>Задаёт имя переменной, в которую будет занесен код причины неудачного трансфера. Возможные значения кодов:</p> <ul style="list-style-type: none"> 16 - занято 22 - нет маршрута 24 - по достижению максимальной длительности вызова <p>Параметры:</p> <ul style="list-style-type: none"> value – имя переменной, vary – имя переменной, хранящей имя переменной.
NumberTranslator	<p>Позволяет задавать правила изменения набранного номера. В блоке может быть несколько экземпляров данного атрибута.</p> <p>Параметры:</p> <ul style="list-style-type: none"> rule – правило изменения набранного номера в формате Billion PBX

	(см. «Billion PBX. Руководство пользователя»).
BillingNumberTranslator	<p>Позволяет задавать правила изменения набранного номера. В блоке может быть несколько экземпляров данного атрибута. Данный атрибут используется, только если включен внутренний биллинг Billion PBX.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • rule – правило изменения набранного номера в формате Billion PBX (см. «Billion PBX. Руководство пользователя»).
AAA	<p>Позволяет задать режим авторизации. Данный атрибут используется, только если включен внутренний биллинг Billion PBX.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • mode – определяет режим авторизации. Возможные значения: <ul style="list-style-type: none"> ○ AAA - аутентификация, авторизация и аккаунтинг, ○ AA - аутентификация и авторизация и ○ A – аккаунтинг.

В приведенном ниже примере вызов перенаправляется на имя, содержащимся в переменной "To".

```
<TransferCall name="Transfer">
<User vary="To"/>
</TransferCall>
```

3.7. Блок EndCall

Блок EndCall завершает вызов.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
sip	Задаёт причину завершения вызова по протоколу SIP. Причина записывается в параметр name .
q931	Задаёт причину завершения вызова по протоколу Q931. Причина записывается в параметр name .

```
<EndCall name="EndCall">
<Sip name="600"/>
<Q931 name="34"/>
</EndCall>
```

Следует обратить внимание, что если IVR-скрипт не содержит вызова данного Блока, звонок переводится на следующий терминатор из маршрутов.

3.8. Блок ParseFolder

Блок ParseFolder перечисляет звуковые файлы и поддиректории для заданного директория. Перечисление начинается при первом обращении к этому блоку и продолжается при последующих обращениях. После исчерпания содержимого директория происходит передача управления на блок, заданный в соответствующем параметре. Данный блок имеет набор дополнительных параметров, приведенных в следующей таблице:

Параметр	Описание
onFoundFile	Задаёт имя блока, которому будет передано управление в случае, когда найден звуковой файл.
onFoundFolder	Задаёт имя блока, которому будет передано управление в случае, когда найдена директория.
onNotFound	Задаёт имя блока, которому будет передано управление в случае, когда исчерпано содержимое исследуемого директория.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Name	Определяет имя директории. Параметры: <ul style="list-style-type: none">• value – имя директории,• vary – имя переменной, хранящей имя директории.
Result	Определяет имя переменной, в которую будет занесено имя файла или директории. Это имя не включает в себя путь и расширение (для звукового файла).

В приведенном ниже примере исследуется содержимое директории *ivr/ru/wav/Directory* со всеми поддиректориями. Результат будет помещён в переменную "ResultName". В случае обнаружения очередного звукового файла управление будет передано на блок "OnFoundFolder", в случае нахождения директории поиск продолжится и в конце поиска управление будет передано на блок "OnEnd"

```
<ParseFolder name="ParseDirectory" onNotFound="OnEnd" onFoundFolder="OnFoundFolder"
onFoundMessage=" ParseDirectory ">
<Name value="Directory" includeSubFolders="True"/>
<Result name="ResultName"/>
</ParseDirectory>
```

3.9. Блок Assign

Блок Assign присваивает значение временным переменным.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Pair	Присваивает значение переменной. В блоке может быть несколько экземпляров данного атрибута. Параметры: <ul style="list-style-type: none">• name – имя переменной,• value – значение переменной,• valueIsKey – допустимые значения <i>True/False</i>. Если значение равно <i>True</i>, значение value берется из переменной по имени "value". По умолчанию – <i>False</i>.

В приведённом ниже примере первый атрибут "Pair" копирует значение переменной "B" в переменную "A". Второй атрибут присваивает переменной "C" значение 100.

```
<Assign name="Assign" next="NextBlock">  
<Pair name="A" value="B" valueIsKey="True" />  
<Pair name="C" value="100" />  
</Assign>
```

3.10. Блок SetLanguage

Блок SetLanguage устанавливает язык IVR.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Language	Задаёт расширение файла языковой локализации. Параметры: <ul style="list-style-type: none">• value – значение (например, "ru" или "en").

```
<SetLanguage name="Russian" next="NextBlock">  
<Language value="ru" />  
</SetLanguage>
```

Для добавления новой локализации необходимо:

1. Перевести файл strings.ru на соответствующий язык и присвоить ему соответствующее расширение.
2. В файл strings.ru добавить соответствующее название языка (например, fr=Французский)
3. Перевести wav файлы, в директорию ivr добавить "соответствующее расширение"/wav и скопировать туда переведённые файлы.

3.11. Блок StoreVariable

Блок StoreVariable сохраняет значение переменной в постоянном хранилище.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Key	<p>Задаёт имя переменной. В блоке может быть несколько экземпляров данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • value – имя переменной, • vary – имя переменной, хранящей имя переменной.
Value	<p>Задаёт значение переменной. В блоке может быть несколько экземпляров данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • value – значение переменной, • vary – имя переменной, хранящей значение переменной.
TTL	<p>Задаёт время жизни переменной. В блоке может быть несколько экземпляров данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • isDate – допустимые значения <i>True/False</i>. Если значение равно <i>True</i>, время жизни берётся как абсолютная величина, в противном случае – со времени последнего использования. По умолчанию – <i>False</i>. • value – время жизни. Примеры использования: <ul style="list-style-type: none"> "5/03/1999 12:34:56" "15/06/1999 12:34:56" "15/06/01 12:34:56 PST" "5/06/02 12:34:56" "5/23/1999 12:34am" "5/23/00 12:34am" "1999/23/04 12:34:56" "Mar 3, 1999 12:34pm" "3 Jul 2004 12:34pm" "12:34:56 5 December 1999" "10 minutes ago" "2 weeks" • vary – имя переменной, хранящей время жизни.

```

<StoreVariable>
<Key value="0"/>
<Key value="billing-user"/>
<Value value="numberFromCell"/>
<TTL value="24:0:0" isDate="False"/>
</StoreVariable>

```

3.12. Блок RetrieveVariable

Блок RetrieveVariable копирует значения переменных из постоянного хранилища во временное. Атрибуты такие же, как и в StoreVariable, только “name” относится к временным переменным, а “value” – к постоянным.

```
<RetrieveVariable>
<Key value="0"/>
<Key value="billing-user"/>
<Value value="numberFromCell"/>
<TTL value="24:0:0" isDate="False"/>
</RetrieveVariable>
```

3.13. Блок ChangeNumber

Блок ChangeNumber изменяет значение числовой переменной.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Var1	Параметры: <ul style="list-style-type: none">• name – имя изменяемой переменной.
Var2	Параметры: <ul style="list-style-type: none">• value – имя изменяющей переменной,• vary – имя переменной, хранящей имя изменяющей переменной.
Action	Определяет действие. Параметры: <ul style="list-style-type: none">• action – допустимые значения: +, -, *, / и % (остаток от деления).

Приведённый ниже пример добавляет к значению переменной “А” значение переменной “В”.

```
<ChangeNumber name="ChangeNumber" next="NextBlock">
<Var1 name="A"/>
<Var2 vary="B"/>
<Action action="+"/>
</ChangeNumber>
```

3.14. Блок SumStrings

Блок SumStrings суммирует значения строковых переменных.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Sum	Определяет переменную, в которую будет собрана итоговая строка.

	<p>Параметры:</p> <ul style="list-style-type: none"> • name – имя переменной.
Item	<p>Определяет строку, присоединяемую к итоговой. В блоке может быть несколько экземпляров данного атрибута.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • value – значение строки или имя переменной, • vary – имя переменной, хранящей значение строки или имя переменной.

В приведённом ниже примере складывается значение переменной “В” со строкой “С” и результат помещается в переменную “А”.

```
<SumStrings name="SumStrings" next=" NextPimitive ">
<Sum name="A"/>
<Item vary="B"/>
<Item value="C"/>
</SumStrings>
```

3.15. Блок CompareStrings

Блок CompareStrings сравнивает значения двух строковых переменных. Данный блок имеет набор дополнительных параметров, приведенных в следующей таблице:

Параметр	Описание
OnEqual	Задаёт имя блока, которому будет передано управление в случае равенства сравниваемых строк.
onNotEqual	Задаёт имя блока, которому будет передано управление в случае неравенства сравниваемых строк.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Var1	<p>Параметры:</p> <ul style="list-style-type: none"> • value – имя первой переменной, • vary – имя переменной, хранящей имя первой переменной.
Var2	<p>Параметры:</p> <ul style="list-style-type: none"> • value – имя второй переменной, • vary – имя переменной, хранящей имя второй переменной.

Caseless	<p>Определяет режим сравнения. Параметры:</p> <ul style="list-style-type: none"> • value – допустимые значения: <ul style="list-style-type: none"> ○ <i>True</i> – учитывается регистр ○ <i>False</i> – регистр не учитывается (по умолчанию).
----------	---

В приведенном ниже примере сравнивается значение переменной “A” с “qwerty”. В случае равенства управление передаётся на Блок **OnEqual**, иначе – на **OnNotEqual**.

```
<CompareStrings name="CompareStrings " onEqual="OnEqual" onNotEqual="OnNotEqual">
<Var1 vary="A" />
<Var2 value="qwerty" />
<Caseless value="TRUE/FALSE"/>
</CompareStrings>
```

3.16. Блок CompareNumbers

Блок CompareNumbers сравнивает значения двух целых переменных. Данный блок имеет набор дополнительных параметров, приведенных в следующей таблице:

Параметр	Описание
OnEqual	Задаёт имя блока, которому будет передано управление в случае равенства сравниваемых переменных.
onGreater	Задаёт имя блока, которому будет передано управление в случае, если Var1 больше Var2.
OnLess	Задаёт имя блока, которому будет передано управление в случае, если Var1 меньше Var2.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Var1	<p>Параметры:</p> <ul style="list-style-type: none"> • value – имя первой переменной, • vary – имя переменной, хранящей имя первой переменной.
Var2	<p>Параметры:</p> <ul style="list-style-type: none"> • value– имя второй переменной, • vary – имя переменной, хранящей имя второй переменной.

```
<CompareNumbers name="Sum" onEqual="..." onGreater="..." onLess="...">
<Var1 vary="A"/>
<Var2 value="B"/>
```

</CompareNumbers>

3.17. Блок Input

Блок Input позволяет ввести dtmf значение, полученное от абонента, в переменную.

Данный блок имеет набор дополнительных параметров, приведенных в следующей таблице:

Параметр	Описание
onFailed	Задаёт имя блока, которому будет передано управление в случае неудачного перевода вызова.
onTryingsExhausted	Задаёт имя блока, которому будет передано управление в случае исчерпания максимального количества попыток допустимого ввода значения.

Атрибуты **Folder**, **Number**, **TelNumber**, **TimeInterval**, **Money**, **Silence**, **Date** и **Time** аналогичны соответствующим атрибутам в блоке [PlayMessages](#). В следующей таблице приводится набор дополнительных атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
File	<p>Задаёт файл для проигрывания. В блоке может быть несколько экземпляров данного атрибута. Параметры:</p> <ul style="list-style-type: none"> • value – имя файла сообщения без расширения, относительно директории /ivr/Russian/wav. Может содержать несколько имен, разделенных пробелами. В этом случае файлы будут проигрываться последовательно. • vary – имя переменной, хранящей имя файла сообщения без расширения. • repeat – количество повторений. По умолчанию 1. Если -1, файл будет проигрываться бесконечно. Если введено не число, предполагается, что это – имя переменной. • tryings – указывает номера попыток, при которых будет проигрываться данное сообщение. Может использоваться как одно, так и несколько значений, например, 1 или 2-4 (со второй по четвертую попытки).
Var	<p>Содержит имя переменной, в которой будет сохранено полученное значение. Параметры:</p> <ul style="list-style-type: none"> • value – имя переменной,

	<ul style="list-style-type: none"> • vary – имя переменной, хранящей имя переменной.
Length	<p>Определяет длину принимаемой dtmf-строки. Параметры:</p> <ul style="list-style-type: none"> • value – длина строки.
MaxTryings	<p>Определяет максимальное количество попыток ввода. Параметры:</p> <ul style="list-style-type: none"> • value – количество попыток ввода.
Terminator	<p>Определяет символ, завершающий строку. Параметры:</p> <ul style="list-style-type: none"> • value – dtmf-символ, например, #.
Timeout	<p>Определяет ограничение на время ввода dtmf-строки абонентом. По умолчанию это ограничение отсутствует. Параметры:</p> <ul style="list-style-type: none"> • value – продолжительность, в секундах.
Default	<p>Определяет значение по умолчанию, которое будет использовано, если не было получено ни одного dtmf-сигнала. Параметры:</p> <ul style="list-style-type: none"> • value – строка символов.

В приведённом ниже примере осуществляется ввод dtmf строки и размещение её в переменной "А", при этом проигрывается сообщение *FileName*. Ввод строки заканчивается, когда пользователь введёт #, или когда пользователь введёт 12-й dtmf, или когда истечёт ограничение на время ввода.

```
<Input name="Input" next=" NextPimitive ">
<File value="FileName" repeat="1"/>
<Var value="A"/>
<Length value="12"/>
<Timeout value="10"/>
<Terminator value="#"/>
</Input>
```

3.18. Блок Choice

Блок Choice позволяет передать управление Блоку в зависимости от введённого dtmf.

Атрибуты **File**, **Folder**, **Number**, **TelNumber**, **TimeInterval**, **Money**, **Silence**, **Date** и **Time** аналогичны соответствующим атрибутам в блоке [PlayMessages](#). В следующей таблице приводится набор дополнительных атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Case	<p>Определяет связь между введённым dtmf и именем Блока для перехода. В блоке может быть несколько экземпляров данного атрибута.</p> <p>Параметры:</p>

	<ul style="list-style-type: none"> • dtmf – значение dtmf-сигнала, • goto – имя блока для перехода в случае ввода указанного сигнала.
Default	<p>Определяет блок для перехода по умолчанию – в случае отсутствия введённых значений, или если эти значения не удовлетворяют ни одному из вариантов, предусмотренных атрибутами Case. Параметры:</p> <ul style="list-style-type: none"> • goto – имя блока для перехода по умолчанию.
Timeout	<p>Определяет ограничение на время ввода dtmf-строки абонентом. По умолчанию это ограничение отсутствует. Параметры:</p> <ul style="list-style-type: none"> • value – продолжительность, в секундах.

В приведенном ниже примере при вводе “1” управление передается на Блок “On1”, при вводе “2” на “On2”, по истечении ограничения времени на ввод – на “Def”. При этом проигрывается сообщение “FileName”. В переменную choice_value записывается значение сделанного выбора. Эта переменная может быть использована в последующих конструкциях.

```
<Choice name="Choice">
<File value="FileName" repeat="1"/>
<Timeout value="10"/>
<Value vary="choice_value"/>
<Case dtmf="1" goto="On1"/>
<Case dtmf="2" goto="On2"/>
<Default goto="Def"/>
</Choice>
```

3.19. Блок Billing

Блок Billing позволяет выполнить запрос на биллинг. Тип биллинга берётся из настроек на странице «Параметры биллинга». Данный блок имеет набор дополнительных параметров, приведенных в следующей таблице:

Параметр	Описание
OnNoBilling	Задаёт имя блока, которому будет передано управление в случае отсутствия связи с биллинг-сервером.
onNoUser	Задаёт имя блока, которому будет передано управление в случае отсутствия пользователя в базе данных биллинг-сервера.
onNoMoney	Задаёт имя блока, которому будет передано управление в случае недостатка средств на счете абонента.
onUserExpired	Задаёт имя блока, которому будет передано управление в случае истечения срока обслуживания абонента.

onBadNumber	Задаёт имя блока, которому будет передано управление в случае неправильно набранного номера.
onUnknown	Задаёт имя блока, которому будет передано управление в случае неопределённой ошибки.
onOk	Задаёт имя блока, которому будет передано управление в случае успешного прохождения проверки биллинг-сервером.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Login	Задаёт имя переменной для логина, с которым будет осуществляться запрос на биллинг. Параметры: <ul style="list-style-type: none"> • value – логин или имя переменной, • vary – имя переменной, хранящей логин или имя переменной.
Number	Задаёт имя переменной для номера телефона, с которым будет осуществляться запрос на биллинг. Параметры: <ul style="list-style-type: none"> • value – номер телефона или имя переменной, • vary – имя переменной, хранящей номер телефона или имя переменной.
Password	Задаёт имя переменной для пароля, с которым будет осуществляться запрос на биллинг. Параметры: <ul style="list-style-type: none"> • value – пароль или имя переменной, • vary – имя переменной, хранящей пароль или имя переменной.
Interval	Описывает переменную, в которую будет заноситься информация о максимальной длительности вызова, полученная с биллинг-сервера.
Money	Описывает переменную, в которую будет заноситься информация об остатке средств на счете абонента, полученная с биллинг-сервера.
MsgFile	Задаёт файл для проигрывания. Параметры:

	<ul style="list-style-type: none"> • value – имя файла сообщения без расширения, • vary – имя переменной, хранящей имя файла сообщения без расширения, • repeat – количество повторений. <p>По умолчанию значение параметра repeat равно 1 (может задаваться переменной). Имя задаётся относительно директория ivr/ru/wav. В имени файла может находиться несколько имён, разделённых пробелами, в этом случае они будут проигрываться последовательно.</p>
NumberTranslator	<p>Позволяет задавать правила изменения набранного номера. В блоке может быть несколько экземпляров данного атрибута.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • rule – правило изменения набранного номера в формате Billion PBX (см. «Billion PBX. Руководство пользователя»).

В приведенном ниже примере производится запрос на биллинг для логина из переменной “login”, пароля из переменной “password” и телефонного номера из переменной “number”. В случае успеха максимальное время вызова в секундах будет занесено в значение переменной “interval”, количество денег на счету у абонента будет занесено в значение переменной “money” и управление передано на Блок “OnOk”. Иначе управление будет передано на Блоки “OnNoBilling” или “OnUserExpired” или на “OnNoUser” или на “OnNoMoney” или “OnBadNumber” или на “OnUnknown” – в зависимости от результата запроса.

```
<Billing name="Billing" next="NextBlock" onNoBilling="OnNoBilling"
onUserExpired="OnUserExpired" onNoUser="OnNoUser" onNoMoney="OnNoMoney"
onBadNumber="OnBadNumber" onUnknown="OnUnknown" OnOk="onOk">
<MsgFile name="FileName" repeat="1"/>
<Login vary="login"/>
<Password vary="password"/>
<Number vary="number"/>
<Interval vary="interval"/>
<Money vary="money"/>
<Currency vary="currency"/>
</Billing>
```

3.20. Блок OS_Shell

Блок OS_Shell позволяет выполнить команду операционной системы или командный файл.

Атрибут	Описание
Command	<p>Задаёт команду операционной системы. Параметры:</p> <ul style="list-style-type: none"> • value – команда операционной системы, • vary – имя переменной, хранящей команду операционной системы.

Result	<p>Получает возвращаемое значение в результате выполнения команды.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • value – имя переменной для хранения возвращаемого значения, • vary – имя переменной, хранящей имя переменной для хранения возвращаемого значения.
--------	---

В приведенном ниже примере блок OS_Shell дожидается исполнения команды и затем переходит к следующему ("Next").

```
<OS_Shell name="OS_Shell" next="Next">
<Command value="c:\exec.bat"/>
<Result vary="result"/>
</OS_Shell>
```

3.21. Блок VoucherIvr

Блок VoucherIvr позволяет абоненту пополнять баланс своего лицевого счета.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Id	<p>Имя ваучера.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • value – имя ваучера, • vary – имя переменной, хранящей имя ваучера.
Customer	<p>Имя клиента внутреннего биллинга.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • value – имя клиента, • vary – имя переменной, хранящей имя клиента.
Result	<p>Сумма, на которую будет увеличен баланс.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • value – сумма, • vary – имя переменной, хранящей сумму.

Рекомендуются следующие варианты применения блока VoucherIvr:

- Позвонить на VoucherIvr как терминатор с номером, равным префиксу для этого терминатора и ID ваучера.

- Послать sms на VoucherIvr как терминатор с номером, равным префиксу для этого терминатора и ID ваучера.
- Послать sms на VoucherIvr как терминатор с номером, равным префиксу для этого терминатора и ID ваучера в тексте sms.

Пример использования блока VoucherIvr:

```
<Voucher name="OnMessage" next="EndCall">  
<Id vary="message-text"/>  
<Customer vary="customer"/>  
<Result vary="result"/>  
</Voucher>
```

3.22. Блок DateSwitchIvr

Блок DateSwitchIvr управляет логикой скрипта в зависимости от параметров даты и времени.

Как и блок IVR, данный блок позволяет включать в себя другие блоки, но только одного типа – **Period**. Каждый включенный экземпляр блока Period описывает некоторый период времени, при попадании в который срабатывает инструкция перехода, заданная в параметре блока **goto**. Например:

```
<Period goto="working">
```

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках блока Period.

Атрибут	Описание
IncludeTime	Определяет интервал времени, включенный в период. Параметры: <ul style="list-style-type: none">• from – время начала,• to – время окончания.
IncludeWeekDay	Определяет интервал дней недели, включенный в период. Параметры: <ul style="list-style-type: none">• from – день начала (0 – воскресенье, 1 – понедельник и т.д.),• to – день окончания.
IncludeMonthDay	Определяет интервал чисел, включенный в период. Параметры: <ul style="list-style-type: none">• from – число начала,• to – число окончания.

IncludeYearDay	<p>Определяет календарный интервал, включенный в период.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • from – дата начала, • to – дата окончания.
ExcludeTime	<p>Определяет интервал времени, исключенный из периода.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • from – время начала, • to – время окончания.
ExcludeWeekDay	<p>Определяет интервал дней недели, исключенный из периода.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • from – день начала (0 – воскресенье, 1 – понедельник и т.д.), • to – день окончания.
ExcludeMonthDay	<p>Определяет интервал чисел, исключенный из периода.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • from – число начала, • to – число окончания.
ExcludeYearDay	<p>Определяет календарный интервал, исключенный из периода.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • from – дата начала, • to – дата окончания.

Приведенный ниже пример описывает следующую логику:

- По умолчанию управление передается Блоку “EndCall”
- <Period goto="working"> - в следующих случаях управление передается Блоку “Working”
 - <IncludeTime from="9" to="18"/> - с 9 до 18
 - <ExcludeTime from="11.00" to="11.15"/> - но не с 11.00 до 11.15

- <ExcludeTime from="14.30" to="15.30"/> - и не с 14.30 до 15.30
- <IncludeWeekDay from="1" to="5"/> - с понедельника по пятницу
- <ExcludeWeekDay from="5" to="5" numberInMonth="4"/> - кроме четвертой пятницы каждого месяца
- <ExcludeMonthDay from="1" to="1"/> - и кроме первого числа каждого месяца
- <ExcludeYearDay from="23.02" to="23.02"/> - и кроме 23-го февраля
- <Period goto="newyear"> - в следующих случаях управление передается Блоку "NewYear"
 - <IncludeYearDay from="1.01" to="1.01"/> - в новый год.

```

<DateSwitch next="EndCall">
<Period goto="working">
<ExcludeTime from="11.00" to="11.15"/>ланч
<ExcludeTime from="14.30" to="15.30"/>обед
<ExcludeWeekDay from="5" to="5" numberInMonth="4"/> четвёртая пятница каждого месяца
<ExcludeMonthDay from="1" to="1"/> первое число каждого месяца
<ExcludeYearDay from="23.02" to="23.02"/> праздник
<IncludeWeekDay from="1" to="5"/>понедельник...пятница
<IncludeTime from="9" to="18"/>рабтчие часы
</Period>
<Period goto="newyear">
<IncludeYearDay from="1.01" to="1.01"/>
</Period>
</DateSwitch>
    
```

3.23. Блок Switch

Блок Switch позволяет передать управление Блоку в зависимости от содержания переменной.

В следующей таблице приводится набор атрибутов, которые могут быть заданы в рамках данного блока.

Атрибут	Описание
Value	<p>Задаёт переменную, значение которой будет анализироваться последующими атрибутами.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • value – имя переменной, • vary – имя переменной, хранящей имя переменной.
Case	<p>Описывает связь между значением переменной и именем блока для перехода. В блоке может быть несколько экземпляров данного атрибута.</p> <p>Параметры:</p>

	<ul style="list-style-type: none"> • value – значение переменной, • goto – имя блока для перехода.
Default	<p>Задаёт блок для перехода в случае несовпадения значений ни с одним атрибутом Case.</p> <p>Параметры:</p> <ul style="list-style-type: none"> • goto – имя блока для перехода.

В приведенном ниже примере для набранного номера “5428518” управление передается на Блок “On1”, для “5428518” на “On2”, для другого номера – на “Def”.

```
<Switch name="Switch">
<Value vary="called-number"/>
<Case value="5428518" goto="On1"/>
<Case value="5428519" goto="On2"/>
<Default goto="Def"/>
</Choice>
```

4. IVR-скрипты, входящие в дистрибутив

В следующей таблице приведены IVR-скрипты, входящие в дистрибутив Billion PBX (файлы имеют расширение **.xml** и находятся в директории **ivr**):

Параметр	Описание
AuthIfNeedAndDest	<p>Если в Billion PBX включен внутренний биллинг, скрипт запрашивает параметры аутентификации. Если такой клиент есть в биллинге, проговаривается его баланс, в противном случае вызов завершается. Далее запрашивается номер, если номер может быть обслужен, то проигрывается максимальное время вызова и осуществляется вызов на этот номер. Если номер не может быть обслужен, то вызов завершается.</p> <p>Если внутренний биллинг выключен, запрашивается номер и совершается вызов на этот номер.</p>
Busy	<p>Проигрывает 10 «коротких гудков» (файл <code>ivr/ru/wav/BusyMessage.wav</code>). Может быть заменен в разделе «Дополнительные параметры» или отредактирован администратором.</p>
CallBack	<p>Скрипт, реализующий функцию обратного вызова. Предназначен для внутреннего использования Billion PBX. Данный скрипт не рекомендуется изменять.</p>
CardPlatform	<p>Скрипт, реализующий функцию работы с телефонными картами. Похож на AuthIfNeedAndDest, но в дополнение позволяет сохранить ПИН код по АОН – чтобы при следующем звонке с этого телефонного номера ПИН код не</p>

	запрашивался. Может быть заменен в параметрах оригинации внешнего устройства или отредактирован администратором.
DateSwitch	Пример использования блока DateSwitch.
EndCall	Завершает вызов. Допускает задание q931 и sip причины завершения вызова. Синтаксис вызова: EndCall.xml;q931=34;sip=600.
MusicOnHold	Бесконечно проигрывает музыку удержания вызова.
Office	Образец IVR-скрипта для офисной АТС. Проигрывает приветствие, предлагает ввести цифры для перехода в отделы или номер для перехода на конкретный телефон. Если ничего не введено, то осуществляется переход на секретаря. При невозможности обслужить вызов осуществляется переход на голосовой почтовый ящик.
OfficeS	Упрощённый вариант Office – после приветствия сразу осуществляется перевод на секретаря.
Play	Проигрывает сообщения и завершает вызов. Синтаксис вызова: Play.xml;wav=ИМЕНА СООБЩЕНИЙ;repeat=КОЛИЧЕСТВО_ПОВТОРОВ. По умолчанию repeat=1. Предназначен для внутреннего использования Billion PBX. Данный скрипт не рекомендуется изменять.
RingBack	Проигрывает бесконечное количество длинных гудков (файл ivr/ru/wav/RingBackMessage.wav). Может быть заменен в разделе «Дополнительные параметры» или отредактирован администратором.
VoiceMailPlay	Проигрывает абоненту оставленные для него сообщения и, по его команде, удаляет их.
VoiceMailStore	Проигрывает приглашение для приема голосовой почты и сохраняет голосовое сообщение. Синтаксис вызова: VoiceMailStore.xml;email=email@host. Если email не задан, то файл не будет посылаться.
VoiceTest	Предлагает записать сообщение и потом проигрывает его звонящему.
Voucher	Предназначен для пополнения счёта.

Billion PBX проигрывает звуковые wav-файлы, указанные в IVR-скрипте. При необходимости файлы, входящие в дистрибутив, могут быть заменены. Более подробно эта тема рассмотрена в Главе «Работа с медиа» документа «Billion PBX. Руководство пользователя».