

Разработка решения для работы с САБ ИРБИС64 в среде PHP Developing solution for implementing IRBIS LAS on the PHP platform

И. С. Батрак
Ассоциация ЭБНИТ,
Москва, Россия

Ivan Batrak
Elnit Association,
Moscow, Russia

В докладе рассматриваются вопросы разработки кроссплатформенного программного обеспечения для работы с САБ ИРБИС64 в среде PHP. Рассмотрен опыт разработки новой реализации интерпретатора форматов САБ ИРБИС64 на основе языков форматирования J-ISIS и CISIS.

Ключевые слова: PHP, язык форматирования pft, unifor, САБ ИРБИС64.

Designing a cross-platform software for implementing IRBIS LAS on the PHP platform is discussed. The new print format language interpreter for IRBIS LAS based on J-ISIS and CISIS formatting language features and capabilities, is also developed.

Keywords: PHP, print format language, unifor, IRBIS LAS.

История проекта

В рамках работы над проектом переноса серверной части САБ ИРБИС64 на платформу Linux возникла идея продолжить наработки, сделанные в рамках проекта WebIrbisNet и сделать кроссплатформенный модуль для веб разработчиков, который бы позволял работать с базами ИРБИС напрямую без участия TCP/IP сервера САБ ИРБИС64.

На сегодняшний день существует проблема, связанная с тем, что современные продукты семейства ИРБИС, такие как ИРБИС 128 и J-ИРБИС не обладают собственным ПО для работы с базами данных САБ ИРБИС64, а также интерпретатором языка форматирования и вынуждены использовать TCP/IP сервер в качестве сервера приложений. Это приводит к удорожанию программных продуктов для конечного потребителя и требует наличие сервера под управлением WINDOWS.

Цель данной разработки предоставить разработчикам на платформе PHP такой инструмент, с помощью которого можно было бы разрабатывать модули сервера приложений для любой ОС с использованием баз данных САБ ИРБИС64, сделать процесс разработки и эксплуатации более удобным и гибким.

В составе САБ ИРБИС64 поставляется динамическая библиотека IRBIS64.DLL, предназначенная для работы с базами ИРБИС, сделанная по образцу ISIS.DLL и есть описание ее интерфейса, однако ее использование в рамках многозадачного приложения, которым является любой веб сервер невозможно. К сожалению, внутреннее устройство IRBIS64.DLL не обеспечивает ни разделения данных между подпроцессами ни синхронизации доступа к общим данным, что неизбежно приводит к аварийному завершению рабочего процесса при наличии параллельно выполняющихся задач.

Внутренне IRBIS64.DLL устроена так, что ей удобно пользоваться в рамках архитектуры CGI, когда каждый запрос клиента выполняется отдельным процессом — обработчиком. Кроме этого, функции в IRBIS64.DLL экспортируются по схеме fastcall, когда при вызове функции аргументы распределяются по регистрам процессора, что не очень удобно использовать. К примеру, для подключения IRBIS64.DLL к среде .NET приходилось создавать вторую DLL, в которой все функции IRBIS64.DLL были заново экспортированы по схеме stdcall, то есть с передачей параметров через стек. Фактически, использование IRBIS64.DLL вне устаревшей архитектуры CGI и приложений написанных в среде Delphi затруднено.

На данный момент ассоциация ЭБНИТ предлагает TCP/IP сервер ИРБИС для OS Linux, что позволяет более гибко подходить к потребностям клиентов и частично снимает вопрос архитектурной зависимости, однако вопрос итоговой стоимости по-прежнему остается.

Ядром системы ИРБИС в целом является язык форматирования ИРБИС. Это специализированный язык программирования, основан на языке PFT (print format language), разработанном для приложения CDS ISIS, предшественника ИРБИС. Представляет собой интерпретируемый сценарий, входными данными, для которого является запись базы данных. На выходе такой сценарий генерирует текст. Используется как для создания печатных форм, так и для формирования поискового индекса баз данных. Долгое время интерпретатор языка форматирования можно было использовать только в приложениях с использованием IRIBIS64.DLL. В рамках проекта WebIrbisNet удалось разработать новый язык форматирования, полностью совместимый с языком форматирования ИРБИС и включающий в себя более современный синтаксис на основе проектов J-ISIS и CISIS. Теперь, чтобы иметь возможность эксплуатировать продукты ИРБИС 128 и J-ИРБИС без обязательного наличия TCP/IP сервера ИРБИС или модуля Web-ИРБИС на разных платформах, в том числе под управлением ОС Linux, понадобилось объединить новый интерпретатор языка форматирования со средой PHP. Новый проект получил название php_irbis64. Разработка ведется в рамках сотрудничества с ассоциацией ЭБНИТ.

Краткая характеристика нового проекта

1) Представляет собой модуль для языка PHP для 32 битных и 64 битных версий. Разрабатывается для последних версий PHP 5 и PHP 7.

2) Разработана версия для ОС Linux, ведется тестовая эксплуатация.

3) Обеспечивает поддержку баз данных как классических версий ИРБИС вплоть до версии 2017, так и новых версий 2018 + и более поздних.

4) Содержит в себе две версии интерпретатора языка форматирования ИРБИС.

Новый язык форматирования xfmt, поддерживающий синтаксис ИРБИС, J-ISIS и CISIS вместе и классический интерпретатор форматов ИРБИС.

Позволяет между ними переключаться во время выполнения.

5) Оба интерпретатора форматов имеют возможность вызывать функции PHP непосредственно из форматов. Это может быть использовано для разработки переносимых функций вместо создания собственных расширений для unifor.

6) Разработан специальный интерфейс для переопределения взаимодействия форматов с базой данных во время выполнения, что может быть востребовано как при использовании модуля для обработки форматов непосредственно на сервере с PHP, так и для использования альтернативных хранилищ данных.

Особенности языка форматирования ИРБИС, которые были также реализованы в новом языке форматирования:

- глобальные переменные
- «висящие» литералы, в том числе после фиктивных команд вывода, которые часто используются в качестве комментариев
- синтаксис комментариев в режиме совместимости с ИРБИС, в других продуктах семейства ISIS синтаксис отличается
 - вывод лишнего повторения в повторяющейся группе
 - контроль работы повторяющихся групп на основе наличия вывода строковых функций &uf, &umargi и S и бесконечный цикл по этой же причине с отсечкой после 5000 итераций
 - остановка повторяющихся групп по команде break
 - измененное поведение функции F
 - пустые блоки if then else
 - синтаксис включения текста внешних форматов поддерживается как в стиле языка форматирования ИРБИС с использованием двух непечатаемых символов, которые видны только в редакторе форматов ИРБИС, так классический синтаксис ISIS в виде @имя_файла. Реализация в

интерпретаторе сделана в стиле J-ISIS. При этом включаемый формат не вставляется как текстовая вставка в вызывающий, как в CISIS и в САБ ИРБИС64, а компилируется и выполняется отдельно. Эта схема является более производительной и позволяет выявлять ошибки во включаемых форматах

- игнорирование переводов каретки при разборе форматов. Реализовано в специальном режиме совместимости с ИРБИС, так как в других языках форматирования перевод каретки является корректным разделителем
- обнуление флагов постредактуры при выполнении любых `unifor`, содержащих вложенный формат
- использования символа восклицательного знака в первой позиции текста формата для индикации кодировки `utf-8`, что пересекается с синтаксисом `escape` литералов языка форматирования CDS ISIS, который также доступен в `xrft` и в самом языке форматирования САБ ИРБИС64. Восклицательный знак в любом месте, кроме первой позиции будет означать по-прежнему команду вывода литерала с символом с ANSI кодом 27 в начале (для отображения `escape` последовательностей). Пример формата (здесь запятая в начале, чтобы отодвинуть восклицательный знак от первой позиции) `!,/я escape литерал/`

В целях совместимости поведения форматов воспроизведены спорные моменты языка форматирования ИРБИС, отличающиеся от языков семейства ISIS и являющиеся частью ожидаемого функционала:

- различное поведение функций `P` и `A` и условных литералов при работе с пустыми подполями
- игнорирование пустых повторений полей, в том числе подавление вывода безусловных литералов и перевода строки
- прерывание выполнения формата после деления на 0
- сброс индекса первого символа подстроки после встречи пустого повторения подполя
- сброс состояние повторяющейся группы в функции `REF`, если та выполняется внутри повторяющейся группы и содержит вложенную повторяющуюся группу

В качестве дополнения к функции `REF`, поведение которой изменено в ИРБИС доделана недокументированная функция `FMT` из состава CISIS. В CISIS функция была сделана аналогично `REF`, только не выводила данные в контекст. В реализации `xrft` данная функция получила возможности оригинальной реализации `REF`, в том числе выполнять свои собственные вложенные повторяющиеся группы. Если нужно выполнить некий независимый код, в том числе внутри повторяющейся группы и использовать `unifor` нет возможности, можно это сделать следующим образом.

Пример формата `(v25, if p(v25) then FMT((| FMT |v25|/FMT|/)) fi, /)`

Ошибка в функционале замены кодов подполей в выходном тексте в режиме данных была исправлена в классическом интерпретаторе. В новом языке форматирования воспроизведение не делалось.

Новое программное решение разрабатывается сразу с учетом портирования в ОС Linux. Учаски кода, зависящие от операционной системы разрабатываются одновременно в двух вариантах.

Язык форматирования

Рабочее название нового языка форматирования `XPFT` (`extended print format language`). Это связано с тем, что синтаксис включает всевозможные варианты синтаксиса языков форматирования CISIS, J-ISIS и ИРБИС. В своей основе реализация языка напоминает структуру языка форматирования J-ISIS.

Интерпретатор языка реализован по схеме `AST` (`abstract syntax tree`). В программном коде определены объекты, представляющие собой реализацию логики всех конструкций языка. При разборе текста формата парсер строит в памяти древовидную структуру объектов, на которую сразу же можно передавать управление. Поддерживается полная потокобезопасность, хотя в данной реализации для PHP это не используется, но откомпилированный формат может быть использован одновременно в нескольких потоках.

Парсер нового языка разработан на основе исходных кодов проекта CISIS. В проекте WebIrbisNet ранее использовался вариант J-ISIS. Позже был разработан новый парсер уже на основе CISIS. Предпочтение варианту на основе CISIS по сравнению с J-ISIS отдано по причине хорошей скорости. Реализация на основе CISIS оказалась гораздо более производительной в основном за счет отсутствия lookahead операций и меньшего количества проверок. Однако, при объединении синтаксиса CISIS и J-ISIS отдельные участки пришлось переписывать с нуля. В некоторых случаях понадобилось все-таки добавить lookahead операции из-за пересечения синтаксиса разных вариантов языка, кроме того в CISIS оказалась совершенно неудачная реализация разбора групп операторов, объединенных в скобки, которая не могла справиться с синтаксисом форматов ИРБИС.

Новый язык форматирования хрft получился менее толерантным к ошибкам синтаксиса, которые часто встречаются в форматах ИРБИС (отсутствующий или лишний блок fi, незакрытые скобки в функциях, отсутствие пробелов между инструкциями). С одной стороны это вызывает некоторые сложности в начале эксплуатации решения, с другой, помогает отлаживать недочеты в существующих форматах и ТВП.

Реализация для платформы РНР получила возможность сообщать пользователю с помощью API РНР ошибки в форматах, чего с классическим интерпретатором ИРБИС делать было нельзя. Особенность нового языка форматирования сообщать ошибки в файлах форматов в с указанием примерного места в файле получилось очень удобной для работы с большим количеством форматов, такими как в каталоге Web-ИРБИС. После проверки форматов последней версии каталога Web-ИРБИС новым интерпретатором был выявлен ряд ошибок синтаксиса в форматах, хотя стоит сказать, что их было значительно меньше, чем в предыдущих версиях.

Ситуация с классическим интерпретатором форматов САБ ИРБИС64 осложняется еще тем, что помимо большого объема файлов, классический интерпретатор не поддерживает многострочные форматы. Эта особенность унаследована в САБ ИРБИС64 от оригинального языка форматирования ISIS, который был взят за основу. В более поздних версиях программных продуктов семейства ISIS поддержка многострочных форматов появилась, но в САБ ИРБИС64 эти новшества не пришли. При обработке файлов производится объединение форматов в одну строку, что увеличивает вероятность трудноуловимых ошибок и делает весьма далекой перспективу вывода отладочных сообщений с привязкой к месту в исходных файлах. При работе с языком форматирования хрft также следует иметь в виду, что часть ошибок выявляются на этапе компиляции, помимо ошибок синтаксиса это и проверка на наличие вложенных повторяющихся групп. В этом случае формат просто не будет выполняться, поэтому следует всегда проверять код возврата функции Irbis_InitPFT. В языке форматирования САБ ИРБИС64 отдельного этапа разбора нет и в таких случаях ошибка выявится во время выполнения, после чего работа прекратится и будет возвращен результат форматирования до момента остановки вместе с кодом ошибки.

Фактически, отладка больших форматов — достаточно трудоемкое занятие и функция нового языка форматирования хрft с отображением ошибок призвана в этом помочь.

Например, следующие две строки внешне синтаксически верные вызовут прерывание выполнения формата и, в случае, с классическим интерпретатором САБ ИРБИС64 разработчику предстоит длительная процедура отладки.

```
if '1' = '1' then 'true' fi
if '2' = '2' then 'also true' fi
```

Здесь ошибка состоит в том, что разработчику всегда следует держать в уме, что перед выполнением формата текст объединяется в одну строку и после объединения получится следующая конструкция.

```
if '1' = '1' then 'true' fiif '2' = '2' then 'also true' fi
```

Во время работы классический интерпретатор САБ ИРБИС64 дойдет до слова fiif и прекратит выполнение, выставив соответствующий код ошибки.

Такое же поведение заложено и в парсер xprt для совместимости с поведением САБ ИРБИС64, хотя в других продуктах семейства ISIS перевод каретки является корректным разделителем. Впрочем, на уровне исходных кодов xprt заложена возможность режим совместимости с САБ ИРБИС64 отключать.

Теперь посмотрим, как на этот формат отреагирует язык xprt. Сохраним данный формат в файл error_test.pft в базе ibis и выполним следующий php сценарий.

```
<pre>
<?php
//инициализация контекста ирбис
$ctx = IrbisAppInit();
//открытие базы
$sp = IrbisInit($ctx, 'ibis');
//включаем интерпретатор xprt
irbis_enable_xprt($ctx,true);
//компиляция формата
$code = Irbis_InitPFT($sp, "@error_test");
if ($code == 0)
{
//загрузка записи 1
IrbisRecord($sp, 0, 1);
 fmt = "";
//выполнение формата
$code = Irbis_Format($sp, 0, 1, $fmt);
echo "$fmt\n";
}
//закрытие базы
IrbisClose($sp);
//закрытие контекста
IrbisAppClose($ctx);
?>
</pre>
```

В результате будет выведен следующий текст.

Warning: Irbis_InitPFT(): @error_test: unexpected token FIF at line 1, column 27 near i if '2' = '2' then 'also tru in C:\xampp_64\htdocs\phptest_doc.php on line 10

При необходимости в реализацию xprt легко добавить информацию о позиции в исходном файле тех или иных блоков для выявления ошибок времени выполнения, таких как деление на 0, бесконечный цикл, запрос отсутствующих файлов или не найденная функция unifor. В САБ ИРБИС64 есть вывод об отсутствующем файле и в версии 2018 в язык форматирования добавлен вывод предупреждения о бесконечном цикле, но это менее информативно, так как конкретное место в файле придется разыскивать, ориентируясь на выходной текст, что не всегда удобно.

Во время выполнения скрипта PHP все откомпилированные форматы находятся в кэше и при последующих обращениях используются повторно. Кэш форматов связан с контекстом приложения, создаваемым функцией IrbisAppInit, после вызова функции IrbisAppClose кэш уничтожается.

Реализация unifor 6 (выполнение внешнего формата) сделана несколько иначе, чем это делается в САБ ИРБИС64. Данная функция используется достаточно часто и многие пользователи замечали, что большие форматы с интенсивным использованием unifor 6 выполняются медленнее. Разработчики САБ ИРБИС64 даже предусмотрели специальный режим, когда вызовы unifor 6 заменяются функцией S с командой включения вложенного формата в текст. Однако, данный режим не учитывает, что имя формата в unifor 6 может быть получено в свою очередь в результате выполнения другого формата и, видимо, поэтому данный режим всегда отключен.

В языке форматирования xpfт применена схема, подсмотренная в J-ISIS и успешно себя зарекомендовавшая в WebIrbisNet. Обе операции включения внешнего формата и вызов unifog 6 выполняются похожим образом. Интерпретатор сначала получает имя формата (в случае с unifog 6 это будет результат выполнения вложенного формата), затем проверяет кэш на наличие уже откомпилированного объекта, если такового нет, файл формата находится по тем же правилам, как это делает САБ ИРБИС64, затем компилируется и помещается в кэш. Кэш существует на время обработки запроса средой РНР. Таким образом манипуляции с файлами и компиляция всегда делаются один раз. Разница поведения в реализации команды с включаемым форматом в xpfт и unifog 6 состоит в следующем. Команда выполнения вложенного формата выполняется в текущем контексте. Для unifog 6 состояние текущего контекста запоминается, а сам контекст очищается.

После выполнения формата, состояние контекста восстанавливается. При этом универсальный счетчик запоминается и после выполнения AST дерева вложенного формата восстанавливается. Затем выполняется постредактура, флаги постредактуры сбрасываются перед началом выполнения и после него, как это делает классический интерпретатор САБ ИРБИС64. Разница в реализации unifog 6 с классическим интерпретатором САБ ИРБИС64 состоит, помимо наличия кэша откомпилированных форматов, в механизме запоминания и восстановления состояния контекста в памяти, при этом не требуется повторное открытие базы ИРБИС и копирования текущей записи в новый контекст, что в совокупности ускоряет процесс.

В классическом интерпретаторе САБ ИРБИС64, несмотря на наличие реализации кэша найденных файлов для unifog 6, применение кэша не предусмотрено, по-видимому в связи с наличием реализации замены фрагментов текста формата на заранее переданные значения в виде @имя_формата#параметр1,параметр2. В языке форматирования xpfт данная опция также реализована и один и тот же файл формата, но с разными значениями параметров будет откомпилирован и внесен в кэш сколько раз, сколько нужно.

В будущем возможно реализовать остальные функции unifog, использующие выполнение вложенных форматов по такой же схеме.

Интересные изменения в синтаксисе языка форматирования

За основу xpfт были взяты языки форматирования J-ISIS и CISIS, которые разрабатывались позже варианта, вошедшего в САБ ИРБИС64, вследствие чего в синтаксисе языка форматирования имеются конструкции, которых нет в форматах ИРБИС. Это числовые и строковые переменные, операторы циклов, также появились дополнительные функции, команды обновления значений полей в стиле CISIS. Вот несколько примеров форматов для иллюстрации. Воспользуемся стандартной базой IBIS из поставки САБ ИРБИС64 2018.

Вывод всего списка результатов поиска с явным указанием базы данных. Здесь знак \$ в конце термина означает, что искомые термины должны начинаться с данной строки. В будущем возможно усовершенствовать функции L и LR и добавить в них поддержку языка запросов CDS ISIS.

```
<pre>
<?php
//инициализация контекста
$ctx = IrbisAppInit();
//открытие базы данных
$sp = IrbisInit($ctx, 'lich');
//включение интерпретатор xpfт
irbis_enable_xpft($ctx, true);

//компиляция формата
Irbis_InitPFT($sp, "ref->IBIS(lr->IBIS(('A=A$')),mfn,v200,/)");
fmt = "";
//выполнение формата
Irbis_Format($sp, 0, 1, $fmt);
//закрытие базы данных
```

```

IrbisClose($sp);
//закрытие контекста
IrbisAppClose($ctx);
echo $fmt;
?>
</pre>

```

Вывод

```
0000000010^АСамоучитель хакера^Еподроб. иллюстрир. руководство^FA. Atsctoy
```

В следующих примерах приведены только изменения относительно предыдущего, формат и результат.

Вывод части списка с указанием диапазона.

Формат.

```
ref->IBIS(lr->IBIS(('A=$'),10,2),mfn,v200,/)
```

Вывод.

```
0000000059^АСправедливость^Екак поступать правильно?^FM. Сэндел ; пер. с англ. А. А. Ка-
линина ; отв. ред. А. Высочкина
```

```
0000000014^AIPhone SDK^Еразработка приложений^FDжонатан Здзиарски ; [пер. с англ.
Александры Маленковой]
```

Вывод списка результатов поиска в цикле.

Здесь нужно немного изменить PHP сценарий.

Нужно заменить строчку с открытием базы данных.

```
$sp = IrbisInit($ctx, 'ibis');
```

Формат.

```
e1:=1,e2:=(npst->('A=A$')), while e1<=e2 (f(e1,1,0),' ->
',ref(lr(('A=A$'),e1,e1),v200) / e1:=(e1+1))
```

Вывод.

```
1 -> 0000000010^АСамоучитель хакера^Еподроб. иллюстрир. руководство^FA. Atsctoy
```

То же самое, с указанием имени базы данных.

```
e1:=1,e2:=(npst->IBIS('A=A$')), while e1<=e2 (f(e1,1,0),' -> ',ref->IBIS(lr->IBIS
(('A=A$'),e1,e1),v200) / e1:=(e1+1))
```

Вывод.

```
1 -> 0000000010^АСамоучитель хакера^Еподроб. иллюстрир. руководство^FA. Atsctoy
```

В данном примере используется усовершенствованный синтаксис, который был ранее разработан в рамках проекта WebIrbisNet. В J-ISIS сборки ноября 2017 это работать не будет. В частности, на момент ноября 2017 года в J-ISIS не поддерживается использование переменных в параметрах, указывающих на диапазон записей функции LR, в функции NPST не предусмотрено явное указание имени базы данных.

Запись, используемая в примерах.

```
//создаем запись на полке 0
```

```
IrbisNewRec($sp, 0);
```

```
//добавляем поля
```

```
IrbisFldAdd($sp, 0, 25, -1, "^a123^aB014809^a456^b123");
```

```
IrbisFldAdd($sp, 0, 25, -1, "^a123^aB014806^a456");
```

```
//компилируем формат  
Irbis_InitPFT($sp, "...");
```

Синтаксис работы с повторениями подполей в том же стиле, что и с повторениями полей.

Формат.

```
(v25[1..2]^a[2..3]*5.5/)
```

Вывод:

```
09456
```

```
06456
```

Прямое присваивание значений полей, подполей, в записи и в глобальных переменных.

Новый синтаксис присваивания значений полей / подполей, реализованный для проекта J-ISIS.
Сделан с поддержкой повторяющихся групп.

Прямое копирование указанных повторений полей.

Формат.

```
g100:=(''),g100:=(v25[1..4]^a[2]), (g100 /)
```

Вывод.

```
V014809
```

```
V014806
```

Копирование в поле результатов выполнения формата

Формат.

```
g100:=(''),g100:=((if p(v25[1..4]^a[2]) then f(occ,0,0) fi,| |v25[1..4]^a[2]|<<<| /)), (g100 /)
```

Вывод.

```
1 V014809<<<
```

```
2 V014806<<<
```

Копирование поля целиком со всеми повторениями.

Формат.

```
v31:=(v25),(v31 /)
```

Вывод.

```
^a123^aV014809^a456^b123
```

```
^a123^aV014806^a456
```

Копирование всех повторений поля с сохранением старого значения.

Формат.

```
v31:=('1 occ'),(v31[2]:=(v25)),(v31 /)
```

Вывод.

```
1 occ
```

```
^a123^aV014809^a456^b123
```

```
^a123^aV014806^a456
```

Копирование всех повторений поля с сохранением старого значения в повторяющейся группе.

Формат.

```
(v31:=(v25)),v31[3]:=('3 occ'),e1:=nocc(v25),(if occ <= e1 then v31[4]:=(v25) fi), (v31 /)
```


Вывод.

```
^a123^aB014809^a456^b123
```

```
^a123^aB014806^a456
```

```
3 occ
```

```
^a123^aB014809^a456^b123
```

```
^a123^aB014806^a456
```

Замена фрагментов подполей

Формат.

```
g100:=(''),g100[3..4]^a[2..3]*2.5:=('12345'/'6789'),g100[2..3]^a[2..3]*2.2:=('12345'/'6789'),(g100 /)
```

Вывод:

```
^a^a12345^a
```

```
^a^a1267895^a
```

```
^a^a6789^a
```

Вызов функций php из языка форматирования.

Данный функционал доступен как для языка форматирования xprt, так и для классического интерпретатора ИРБИС.

```
//функция php для использования в формате
```

```
//параметры
```

```
//$sp – открытая база данных ИРБИС
```

```
//$Curr_Shelf – номер полки, где текущая запись
```

```
//$Occ_Exit – текущее повторение
```

```
//$params – строка с параметрами
```

```
function php_unifor($sp, $Curr_Shelf, $OCC_Exit, $params)
```

```
{
```

```
    $objDateTime = new DateTime('NOW');
```

```
    return trim($params)." current date is ".$objDateTime->format('c');
```

```
};
```

```
$callbacks = array
```

```
(
```

```
    "PHP_UF" => "php_unifor"
```

```
);
```

```
Irbis_InitPFT($sp, "v100[1] := (&php_uf(' +++ ')), v100[2] := (&php_uf(' --- ')),  
(v100 /)");
```

Вывод:

```
+++ current date is 2019-05-22T02:50:07+00:00
```

```
--- current date is 2019-05-22T02:50:07+00:00
```

Отладка ошибок в реальных форматах

Для примера также воспользуемся базой IBIS из демонстрационной версии САБ ИРБИС64 2018. В данном тесте путь к Deposit настроен на каталог Web-ИРБИС 2018. Каталог Web-ИРБИС был загружен в начале 2019 года, все ошибки, которые были выявлены языком xprt были переданы разработчику. Возможно, в текущих версиях они уже исправлены. Выполним следующий сценарий PHP.

```
<pre>
```

```
<?php
```

```
//инициализация контекста
```

```

$ctx = IrbisAppInit();
//открытие базы данных
$sp = IrbisInit($ctx, 'ibis');
//включение интерпретатор xpft
irbis_enable_xpft($ctx, true);

//загрузка записи 1
IrbisRecord($sp, 0, 1);
//компиляция формата fullwebr
Irbis_InitPFT($sp, "@fullwebr");
$fmt = "";
//выполнение формата
Irbis_Format($sp, 0, 1, $fmt, $callbacks);
//закрытие базы данных
IrbisClose($sp);
//закрытие контекста
IrbisAppClose($ctx);
echo $fmt;
?>
</pre>

```

Вывод.

```

Warning: Irbis_Format(): @oant_H: FI expected at line 142, column 54 near fi fi)
" <br> <b> ~~Примеча in C:\Apache24\htdocs\phptest_jisis.php on line 34

```

```
<--MFNCOUNT-->.
```

004.4

К 60

Колисниченко, Денис Николаевич.

Android для пользователя [Текст] : полезные программы и советы / Денис Колисниченко. - Санкт-Петербург : БХВ-Петербург, 2013. - 254 с. : ил + 24 см. - ISBN 978-5-9775-06 80-9 : Б. ц. Указ.

Здесь при выводе фрагмента текста в примерном месте ошибки текст выводится с проверкой на многобайтное представление символов в кодировке utf8, чтобы не выводились поврежденные символы.

Найдем формат oant_H.pft и найдем в нем строку 142.

Строка выглядит вот так.

```
if &unifor('К117а2.mnu|'v117^2)=' then v117^2)fi fi)
```

Здесь рядом с fi fi мы видим лишнюю скобку, поправим строку вот так.

```
if &unifor('К117а2.mnu|'v117^2)=' then v117^2 fi fi)
```

Выполняем заново, ошибок — нет.

Вывод.

```
<--MFNCOUNT-->.
```

004.4

К 60

Колисниченко, Денис Николаевич.

Android для пользователя [Текст] : полезные программы и советы / Денис Колисниченко. - Санкт-Петербург : БХВ-Петербург, 2013. - 254 с. : ил + 24 см. - ISBN 978-5-9775-06 80-9 : Б. ц. Указ.

~~УДК~~ 004.4

~~БК~~ 3884.16-56-05,0 + 3973.233-018.2,0

Рубрики: Радиоэлектроника--Электрическая связь--Радиосвязь--Радиотелефонная связь, Мобильная связь--Радиотелефоны, Вычислительная техника--Вычислительные машины электронные цифровые--Автоматическая обработка информации, ANDROID, операционная система--Программирование

Экземпляры
Учебная литература

(для доступа требуется авторизация)

Итог

Платформа ИРБИС аккумулирует передовой опыт в разработке библиотечных решений, является простой и надежной в эксплуатации, а также недорогой системой. Не требует участия высокооплачиваемых специалистов. Предоставляет расширенные возможности к конфигурации. К недостаткам можно отнести недостаточную развитость средств разработки и до недавнего времени отсутствие кроссплатформенных серверных решений. Данная разработка призвана объединить вместе сообщество разработчиков PHP и ИРБИС, повысить гибкость и конкурентоспособность современных продуктов семейства ИРБИС. В ближайшей перспективе видится завершение тестирования и выпуск версии решения для ОС Linux, дальнейшее развитие функционала. Есть идеи реализации альтернативных хранилищ данных в различных СУБД, обновления реализации внутренней архитектуры приложения, чтобы сделать ее ближе к оказавшейся довольно удачной архитектуре J-ISIS. В завершении хотелось бы выразить надежду на выпуск данного программного продукта официально под эгидой ассоциации ЭБНИТ. Помимо решения для PHP выпуск SDK для работы с САБ ИРБИС64 для платформ Java, .NET, Python также может быть интересен пользователям. На данный момент помимо версий для ОС Windows подготовлены сборки модуля для PHP 5.6 в Linux Rosa Fresh R10 и для PHP 7.3 в Linux Debian 9. Простейший тестовый сценарий PHP, который отображает результат выполнения формат fullwebr.pft для записей базы IBIS из состава САБ ИРБИС64 2018 можно посмотреть на сервере ООО «Ирбис-Консультант» по адресу http://185.238.136.127/php_irbis64/fullwebr.php