

STM Developer Suite

- ▶ Сервер телеметрии
Master 17

Руководство пользователя



Содержание

Назначение программы	1-6
Системные требования	1-6
Установка и запуск программы	1-7
Командная строка. Описание ключей	1-7
Контроль работоспособности	1-10
Структура файла настройки вывода отладочной информации	1-11
Примеры сообщений в лог-файле	1-13
Завершение программы	1-14

Сервер телеметрии Master17

Назначение программы

Сервер телеметрии (Master17) является программным компонентом комплекса Net Server TM, предназначенным для контроля, хранения, модификации, и выдачи по запросу пользователя данных. Он имеет язык сценариев (скриптов), организованный

в виде блоков скриптов, который реализует необходимые пользователю алгоритмы контроля и управления состоянием контролируемых объектов и телеуправление

реальными объектами контроля.

Сервер телеметрии выполняет следующие задачи:

- ◇ создание базы данных, чтение \ запись данных в нее
- ◇ контроль целостности базы данных, автоматическое (по условиям времени или дополнительным условиям) создание резервных копий базы данных
- ◇ запуск интерпретатора языка TCL и выполнение скриптов
- ◇ создание серверного TCP/IP сокета для организации связи с клиентскими приложениями по протоколу STM 1.5

Системные требования

Для работы сервера телеметрии требуется IBM-совместимый компьютер с установленной на нём операционной системой Windows, Linux, FreeBSD, Solaris. В операционной системе должен быть установлен интерпретатор языка Tcl\Tk Tclsh версии не ниже 8.4.4.

Требования к системным ресурсам определяются, в основном, размером базы данных, которая будет создана сервером телеметрии. Свободное место на жестком диске компьютера должно быть не меньше, чем «Размер_БД» + 30Мбайт. Объем оперативной памяти должен быть не меньше 0,5 Гбайт, желательно 2 Гбайта или больше, так как при работе с файлами базы данных, сервер телеметрии использует механизм отражения содержимого файлов в память процесса (мапирование). При нехватке оперативной памяти (объем БД больше объема ОП), операционная система задействует свободный объем жесткого диска (swar файлы), что замедляет скорость работы сервера телеметрии.

Для успешного функционирования программы необходим файл конфигурации (*.stm), который создается специализированным программным модулем «Редактор конфигураций StmConf» комплекса «Net Server TM».

Установка и запуск программы

Программа не требует установки, достаточно скопировать исполняемый файл в некоторую (обычно /stm или c:\stm) директорию, по желанию пользователя.

Программа должна запускаться пользователем с правами администратора (root).

Для успешного функционирования программы при запуске в командной строке должно быть указано имя файла конфигурации (*.stm). Из этого файла сервер телеметрии извлекает информацию о базах данных, которые должны быть созданы;

настройках соединения с программами – клиентами; в файле конфигурации содержатся скрипты, которые выполняются в процессе работы Master-а и пр. Если имя конфигурации не было указано, Master17 завершает работу. Файл конфигурации должен находиться в той же директории, что и исполняемый файл сервера телеметрии.

Перед запуском сервера необходимо создать:

- в директории, в которой находится исполняемый файл Master-а, поддиректорию, в которой будут находиться базы данных. Имя поддиректории задается при создании файла конфигурации (см. документ «Редактор конфигураций StmConf Руководство пользователя»). Далее данную поддиректорию будем именовать директория БД.
- в директории, в которой находится исполняемый файл Master-а, необходимо создать файл настроек вывода отладочной информации (*.conf) в лог файл (*.log). Имя файла задается при создании файла конфигурации. Формат и содержимое файла описаны в пункте 3.2. При отсутствии файла настроек вывода отладочной информации, сервер завершает работу.

Для запуска сервера телеметрии рекомендуется использовать командный скрипт «start», находящийся в рабочей директории сервера телеметрии. Содержимое скрипта описано в документе «Серверная часть комплекса "Net Server TM". Руководство системного программиста».

Командная строка. Описание ключей

Командная строка имеет вид:

```
Master17 [ключи] <имя_файла_конфигурации>.stm
```

В случае отсутствия параметров в командной строке сервер завершает работу, предварительно выдав в консоль правила использования программы.

В командной строке могут быть указан(ы) ключ(и), с помощью которых мож-

но управлять некоторыми особенностями функционирования программы. Ключи разделяются пробелами. Перед ключом должен находиться символ '\-'. Имя ключа – прописная латинская буква. Непосредственно за ключом может идти некоторое значение.

Пример:

```
Master17 -a -v123456789 -115000 <имя_файла_конфигурации>.stm
```

Подробно опишем существующие в настоящий момент ключи:

- **<v<целое_беззнаковое>**, значение по умолчанию – нет, означает имя поддиректории в директории БД, в которой находится либо последняя версия БД, обрзовавшаяся при остановке Master17, либо сохраненная ранее версия (архив).

Указывает, что при старте программы в создаваемую базу данных необходимо скопировать данные из указанной версии базы данных.

Подробнее об этом.

В директории БД сервером телеметрии создаются поддиректории, в которых находятся непосредственно файлы БД. Автоматическое создание новой поддиректории происходит каждый раз после изменения файла конфигурации (*.stm), это своего рода история изменений БД.

Если при изменении файла конфигурации старые данные не нужны, необходимо остановить Master17, скопировать измененный файл конфигурации в рабочую директорию сервера, запустить сервер без данного ключа. При этом старая БД останется, а новая БД будет создана в новой поддиректории в директории БД.

Если же данные из старой БД (все или частично) нужны, необходимо остановить сервер, скопировать новый файл конфигурации и запустить мастер с данным ключом (тот же эффект можно достигнуть просто скопировав файл конфигурации без остановки сервера). При этом, значения типов \ объектов, которые присутствуют в обеих версиях файла конфигурации, будут из старой БД скопированы в новую.

Имена поддиректорий генерируются автоматически на основе времени последнего исправления файла конфигурации, поэтому они представляют из себя целое беззнаковое число, переведенное в строковой вид.

Пример:

Пусть рабочая директория сервера - /stm, имя директории БД - /test_db. При работе сервера была создана поддиректория с файлами БД /12345555, при этом полный путь к файлам БД будет - /stm/test_db/12345555. Мы изменили файл конфигурации *.stm, добавив новый тип с некоторыми объектами и добавили (удалили) некоторые объекты из существовавших ранее типов. Теперь нам нужно остановить Master17, скопировать файл конфигурации и запустить сервер с ключом -v12345555. При этом будет создана новая поддиректория с файлами БД по

полному пути /stm/test_db/12346666. И в новую БД будут частично скопированы данные из старой БД, причем те типы \ объекты, которые отсутствовали с старой версии БД будут «пустые», а типы \ объекты, которые присутствуют в обеих версиях конфигурации, будут «заполнены».

- **l**<целое беззнаковое>, значение по умолчанию – 3500.

Ограничение очереди (в пакетах) на прием \ передачу от \ к каждого клиента сервера телеметрии, при обмене данными по протоколу STM1.5. Возможность такого ограничения введена для распределения нагрузки на сервер телеметрии при наличии большого числа клиентских соединений.

- **s**<целое беззнаковое>, значение по умолчанию – 500

Дополнение к ключу l. Указывает, какое максимальное количество пакетов сервер может отправить клиенту за один раз. Рекомендуется использовать ключ при соединения с клиентами по медленным \ подверженным помехам линиям, при этом нужно снизить значение по отношению к значению по умолчанию.

- **a**<целое беззнаковое в диапазоне 0...23>, значение по умолчанию – 1

Указывает на необходимость автоматического ежедневного создания архивов текущей БД. При отсутствии ключа архивирование не происходит. Указанное значение – час в сутках, при наступлении которого процесс архивирования будет запускаться.

Рекомендуется запускать архивирование в часы наименьшей нагрузки на сервер телеметрии, например, ночью. При указании ключа без значения – архивирование будет запускаться в час ночи.

Файлы архивов создаются в директории, которая автоматически создается в случае ее отсутствия:

<рабочая директория сервера >\backup\<имя конфигурационного файла>\auto.

Например: stm\backup\test_1\auto или c:\master\backup\test_1\auto.

Имя архива: `год_полностью`"месяц`"день`"час`"минуты`"секунды`.tar, причем время на основе которого формируется имя файла - это время начала процесса архивирования. Например, имя архива может быть таким - 20071205010001.tar.

Для архивирования необходимо:

- права процесса на создание папок в рабочей директории;
- архиваторы tar и gzip должны быть в рабочей директории сервера.

В директории хранятся архивы только за 7 предыдущих суток, более старые версии архивов автоматически удаляются.

- **b**<целое беззнаковое в диапазоне 10...180>, значение по умолчанию – 30, означает период (в сутках) между архивированием не удаляемых автоматически копий БД. Указывает на необходимость автоматического периодического создания архивов текущей БД. При отсутствии ключа архивирование не происходит.

Время запуска архивирования берется по значениям и правилам ключа "а".

Файлы архивов создаются в директории, которая автоматически создается в случае ее отсутствия:

<рабочая директория сервера >\backup\<<имя конфигурационного файла>\log.

Например: stm\backup\test_1\log или c:\master\backup\test_1\log.

Имя архива формируется так же, как и для ключа «а», требования те же.

Архивы данных автоматически не удаляются.

- **r**, значений нет

Указывает, что мастер телеметрии запускается в режиме «READ ONLY» для баз данных. Ключ используется при "просмотре" когда-то заархивированных баз. Можно использовать вместе с ключом «v».

- **c**, значений нет

Предназначен для использования во время восстановительных работ при сбое баз данных. Ключ может использовать только специально обученный персонал. Использовать только вместе с ключом «v» и только в ручном режиме работы с сервером. Использовать ключ в скриптах (например, скриптах модуля **Spy**) категорически запрещается.

При использовании данного ключа указанная (ключ «v») БД просто копируется в новую поддиректорию в директории БД. При этом структура старой БД не разбирается.

Контроль работоспособности программы

Контроль работоспособности сервера телеметрии осуществляется с помощью лог файла (*.log). Типы информации, которые выводятся в лог файл, задаются с помощью файла настройки вывода отладочной информации (*.conf). Файлы находятся в рабочей директории сервера.

При старте сервера телеметрии содержимое *.log файла удаляется. Во время работы сервера *.log файл очищается через каждые 50000 записанных сообщений, при этом максимальный размер лог файла обычно не превышает 10Мбайт.

Структура файла настройки вывода отладочной информации в лог файл `strm.conf`

Этот файл представляет собой набор ключей, которые регламентируют запись тех или иных данных в лог файл. При изменении состояния в режимы On и Off (включен и выключен соответственно) появляется возможность получать только ту информацию, которая необходима.

Менять состояние ключей допускается без завершения программы, программа автоматически проверяет изменение содержимого файла настроек каждые 20 секунд.

Не рекомендуется без необходимости включать максимальную отладку, т.к. в этом случае работа программы несколько замедляется из-за интенсивной работы с файловой системой. Имена файлов задаются на этапе создания \ редактирования файла конфигурации.

Описание ключей:

Общие ключи:

- **DiagEnable** - разрешает или запрещает вывод в лог-файл диагностики вообще. При установке данного ключа в «Off» вывод информации в лог файл прекращается, несмотря на состояния других ключей.
- **ShowNum** - разрешает или запрещает вывод в лог-файл порядкового номера отладочного сообщения. Нумерация начинается по старту сервера, при очистке содержимого *.log файла нумерация не сбрасывается.
- **ShowTime** - разрешает или запрещает вывод в лог-файл времени события.
- **ShowSrcFile** - разрешает или запрещает вывод в лог-файл имени файла с исходным кодом программы. Ключ актуален для сертифицированных пользователей.

Ключи по типам информации:

- **AllErrorMsg** - разрешает или запрещает вывод в лог-файл информации о различных ошибках, возникших при работе сервера. Например, ошибки при работе с файловой системой, ошибки выполнения Tcl-скриптов и др.
- **AllWarningMsg** - разрешает или запрещает вывод в лог-файл информации о некритичных ошибках, возникших при работе сервера. Например, сообщения о некорректных запросах клиентов сервера, запросах от клиентов или Tcl-скриптов к несуществующим данным и т.п.
- **AllUserMsg** - разрешает или запрещает вывод в лог-файл сообщений о начале/окончании этапов работы сервера. Например, старт \ завершение программы, запуск интерпретатора языка Tcl\Tk и т.п.
- **AllHardMsg** - не используется.

- `AllNetMessage` - разрешает или запрещает вывод в лог-файл подробную информацию о работе с сетью (сокетные соединения, протокол STM 1.5).
- `AllDbMsg` - разрешает или запрещает вывод в лог-файл подробную информацию о работе с базами данных.
- `AllEventMsg` - разрешает или запрещает вывод в лог-файл подробную информацию об обработке событий (скрипты TCL).
- `AllLoadMsg` - разрешает или запрещает вывод в лог-файл сообщений о загрузке (разбор файла конфигурации).
- `AllLinkMsg` - разрешает или запрещает вывод в лог-файл сообщений об установлении внутренних связей между блоками данных, переменных сервера.
- `AllStepMsg` - разрешает или запрещает вывод в лог-файл временных отладочных сообщений. В режиме штатной работы сервера включать этот ключ не рекомендуется.

При стабильной работе сервера телеметрии рекомендуется установить только следующие ключи: `AllErrorMsg`, `AllWarningMsg` и `AllUserMsg`.

Пример файла:

```
On   DiagEnable
On   ShowNum
On   ShowTime
Off  ShowSrcFile
On   AllErrorMsg
On   AllWarningMsg
On   AllUserMsg
Off  AllHardMsg
Off  AllNetMessage
Off  AllDbMsg
Off  AllEventMsg
Off  AllLoadMsg
Off  AllLinkMsg
Off  AllStepMsg
```

Необходимо отметить, что в лог файл может выводиться отладочная информация из TCL-скиптов, запускаемых сервером телеметрии. И лог файл, и файл настройки вывода отладочной информации скриптами используются те же, что и сервером. При этом в файле настройки скриптами используются другие ключи, не описанные выше; информация о том, какие именно ключи можно \ нужно добавить находится в соответствующей документации на TCL-скрипты.

Примеры сообщений в лог файле

Вывод информации в лог файл сервер телеметрии начинается сразу после запуска, еще не прочитав содержимое файла конфигурации *.stm. Поэтому сообщения о первых этапах выполнения программы записываются в файл «debug.log», создающийся в рабочей директории сервера. После разбора первой части файла конфигурации сервер получает информацию об имени лог файла, и сообщения начинают выводиться в указанный лог файл по правилам, описанным в файле настройки.

Большинство отладочных сообщений, записываемых в лог файл, предназначено для программистов и системных программистов, работающих с сервером телеметрии.

Ниже будут описаны сообщения, записываемые в лог файл при установке ключей `AllErrorMsg`, `AllWarningMsg` и `AllUserMsg` и при корректной работе сервера телеметрии. Сообщения об ошибках, отладочные сообщения подробно описаны документе «Сервер телеметрии Master17. Руководство системного программиста».

Содержимое файла «debug.log» при старте сервера телеметрии удаляется.

Типовое содержимое файла «debug.log»:

```
1 [08.08.2007 16:40:55.130] [main.cpp:68] *** Начало выполнения 0
2 [08.08.2007 16:40:55.130] [GlobalCfg.cpp:217] LOAD Старая конфигурация 0
3 [08.08.2007 16:40:55.130] [GlobalCfg.cpp:218] LOAD Текущая конфигурация 1186546821
4 [08.08.2007 16:40:55.130] [SrvCfgStruct.cpp:93] LOAD_CONF Начало разбора конфигурации
```

Строка 1 – собственно, момент старта сервера.

Строки 2, 3 – информация об известных файлах конфигурации (*.stm). Как видно из строки 2 – о старой конфигурации данных нет, число в конце строки 3 – время последней модификации файла конфигурации, именно такое имя (1186546821) будет иметь поддиректория с файлами базы данных в директории БД.

Строка 4 – сообщение о начале разбора конфигурации.

Содержимое лог файла:

```
1 [08.08.2007 16:40:55.140] MESSAGE Открываем серверный сокет протокола STM1 на порту 9999
2 [08.08.2007 16:40:55.150] MESSAGE Tcl DebugInit - Ok
3 [08.08.2007 16:40:55.359] Начало цикла
```

Сервер телеметрии Master 17

```
4 [08.08.2007 18:14:53.339] *** SetStop arg=15
5 [08.08.2007 18:14:53.339] Окончание выполнения 15
6 [08.08.2007 18:14:53.339] STOP Событие SrvSockSTM15=0x80cc2f8
7 [08.08.2007 18:14:53.339] MESSAGE Закрываем серверный сокет
протокола STM1
```

Строки 1, 6, 7 – создание (успешное) и закрытие серверного TCP/IP сокета для приема входящих соединений от клиентского программного обеспечения.

Строка 2 – завершение (успешное) запуска интерпретатора Tclsh и Tcl-скриптов

Строка 3 – сообщение о переходе сервера в событийный режим

Строки 4, 5 – сообщения о получении от операционной системы сигнала на завершение работы

Завершение программы

Каких-либо определённых требований к закрытию программы нет, поэтому процесс завершения работы STRM идентичен процессу закрытия любого приложения для конкретной операционной системы.

Для удобства пользователей рекомендуется использовать командный скрипт «stop», находящийся в рабочей директории менеджера потоков. Содержимое скрипта описано в документе «Серверная часть комплекса "Net Server TM". Руководство системного программиста».

