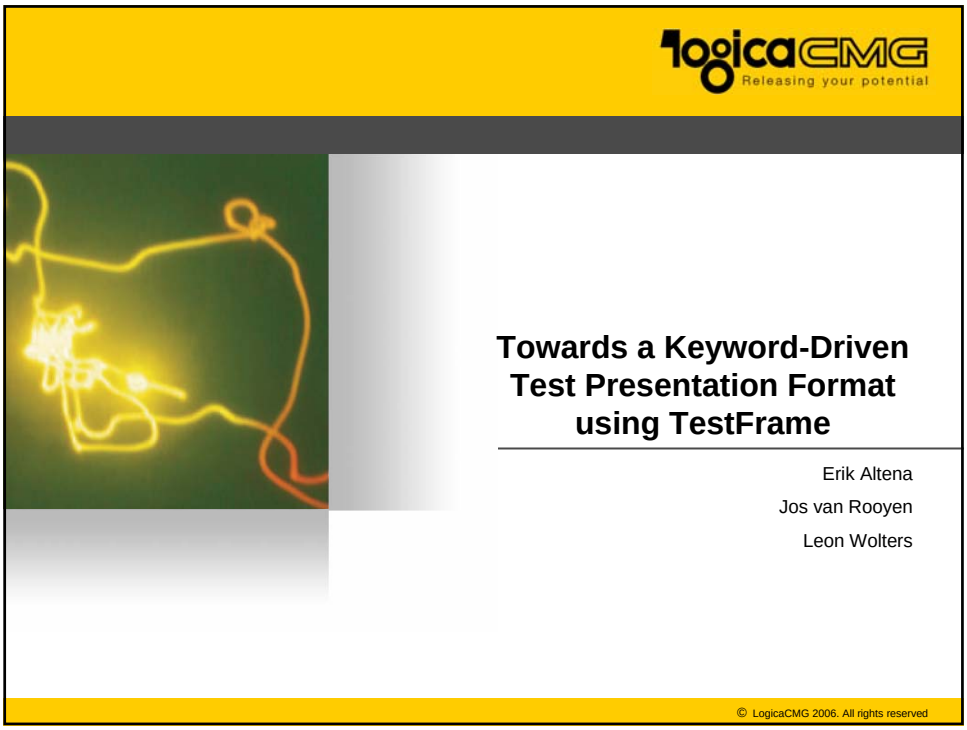
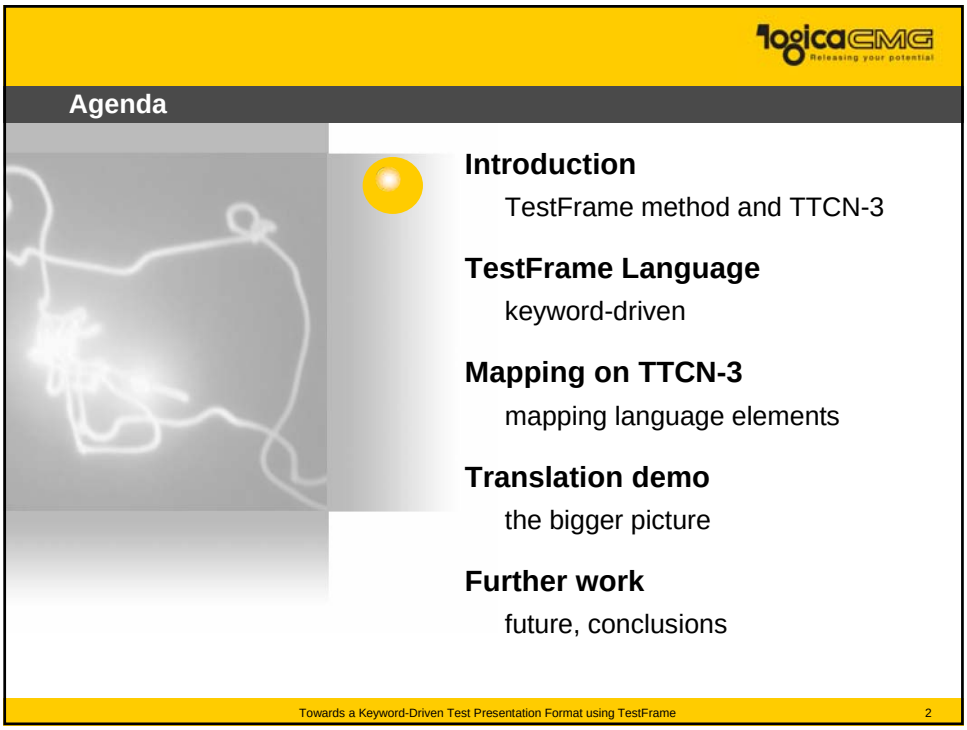
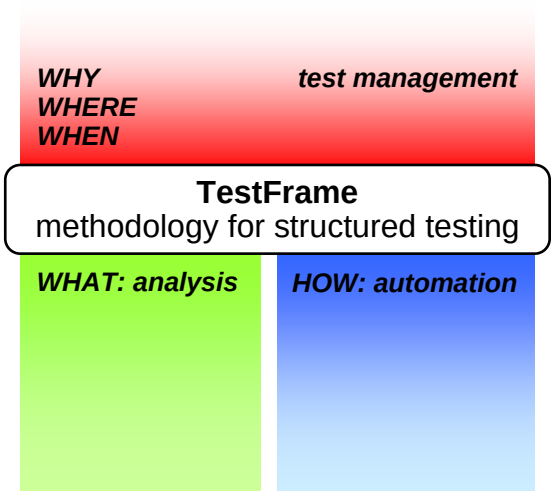

Releasing your potential
Towards a Keyword-Driven Test Presentation Format using TestFrame
Erik Altena
Jos van Rooyen
Leon Wolters
© LogicaCMG 2006. All rights reserved

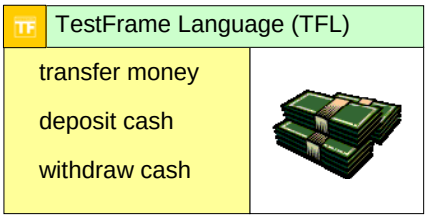

Releasing your potential
Agenda
Introduction
TestFrame method and TTCN-3
TestFrame Language
keyword-driven
Mapping on TTCN-3
mapping language elements
Translation demo
the bigger picture
Further work
future, conclusions
Towards a Keyword-Driven Test Presentation Format using TestFrame
2

Introduction



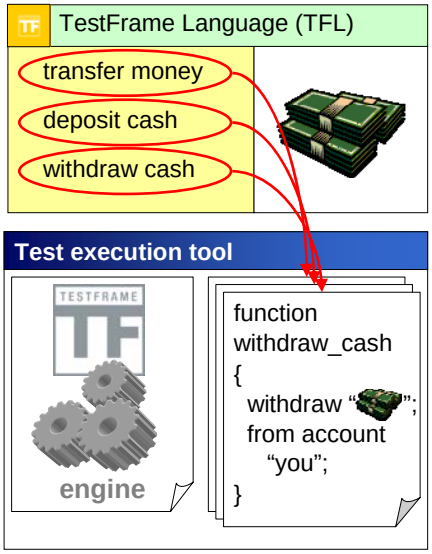
- TestFrame is a methodology for structured software testing.
- Its basic premise is the separation between *what* to test and *how* to test it
- Test analysis (what) is separated from its technical implementation (how)

Introduction



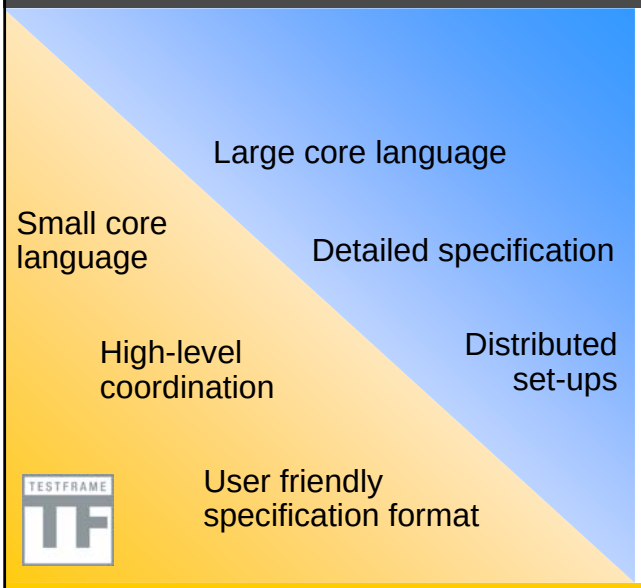
- Test analysis focuses on the test logic; specified with keywords.
- Written down in its own TestFrame Language (TFL).
- Keywords can be reused with different test data.

Introduction



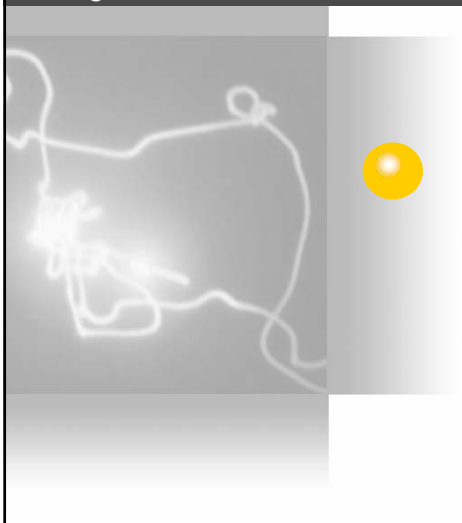
- TFL is a test tool independent test language.
- During test automation each keyword receives its technical implementation in the chosen test tool.
- Processing requires an off-the-shelf or tool-specific TestFrame Engine

Introduction



- TestFrame and TTCN-3 are both used for test specification.
- Each has its strengths…
  - …that suit some domains better than others.
- When combined they will appeal to more types of testers from more domains.

Agenda



Introduction
TestFrame method and TTCN-3

TestFrame Language
keyword-driven

Mapping on TTCN-3
mapping language elements

Translation demo
the bigger picture

Further work
future, conclusions

TestFrame Language

cluster	TFL_basics
author	Erik Altena
version	1.0
start application	Application Home Bank
log in	User EA1234
test case	TFB_1
get balance	Balance &Keep(start
transfer money	To account 123.45.678
transfer money	234.56.789
check balance	Balance &startbal - 6
log out	
	Close the a
stop application	Application Home Bank

cluster	TFL_basics		
author	Erik Altena		
version	1.0		
	Application		
start application	Home Bank		
	User	Password	
log in	EA1234	welcome	
test case	TFB_1		
	Balance		
get balance	&Keep(start_bal)		
	To account	Amount	Description
transfer money	123.45.678	25,00	Borrowed
transfer money	234.56.789	653,88	Rent
	Balance		
check balance	&startbal - 678,88		
log out			
	Close the application manually after failure!		
	Application		
stop application	Home Bank		

TestFrame test scripts are usually written in a spreadsheet program.

These sheets are translated to tab-delimited text files, called clusters.

Clusters are written in TestFrame Language (TFL).

TestFrame Language

cluster	TFL_basics		
author	Erik Altena		
version	1.0		
	Application		
start application	Home Bank		
	User	Password	
log in	EA1234	welcome	
test case	TFB_1		
	Balance		
get balance	&Keep(start_bal)		
	To account	Amount	Description
transfer money	123.45.678	25,00	Borrowed
transfer money	234.56.789	653,88	Rent
	Balance		
check balance	&startbal - 678,88		
log out			
	Close the application manually after failure!		
	Application		
stop application	Home Bank		

- Each test line starts with a keyword.
- All following fields contain parameters.
- Test lines without a keyword are ignored or used for parameter names.

TestFrame Language

cluster	TFL_basics		
author	Erik Altena		
version	1.0		
	Application		
start application	Home Bank		
	User	Password	
log in	EA1234	welcome	
test case	TFB_1		
	Balance		
get balance	&Keep(start_bal)		
	To account	Amount	Description
transfer money	123.45.678	25,00	Borrowed
transfer money	234.56.789	653,88	Rent
	Balance		
check balance	&startbal - 678,88		
log out			
	Close the application manually after failure!		
	Application		
stop application	Home Bank		

reporting

user-defined

structuring

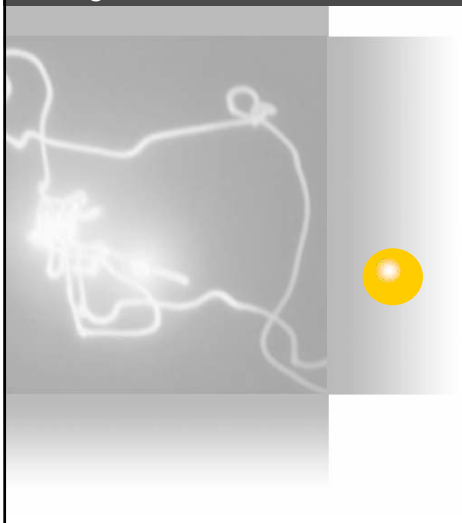
user-defined

Pre-defined keywords for test control:

- reporting
- structuring
- calling subclusters
- flow-of-control
- data transfer, etc.

Keywords that are user-defined describe the test procedure.

Agenda



Introduction
TestFrame method and TTCN-3

TestFrame Language
keyword-driven

Mapping on TTCN-3
mapping language elements

Translation demo
the bigger picture

Further work
future, conclusions

Mapping on TTCN-3

cluster	TFL_basics
author	Erik Altena
version	1.0
start application	Application Home Bank
log in	User EA1234
test case	TFB_1
get balance	Balance &Keep(start
transfer money	To account 123.45.678
transfer money	234.56.789
check balance	Balance &startbal - 6
log out	
	Close the a
stop application	Application Home Bank

```
module TFL_basics {
        import from Keywords all;

        testcase TFB_1() {
                var  float start_bal;

                get_balance(start_bal);
                transfer_money("123.45.678", "25,00", "Borrowed);
                transfer_money("234.56.789", "653,88", "Rent");
                check_balance(start_bal – 678,88);
                log_out();
                stop_application("Home Bank");
        }

        control () {
                tfl_header_cluster("TFL_basics");
                tfl_header_author("Erik Altena");
                tfl_header_version("1.0");
                start_application("Home Bank");
                log_in("EA1234", "welcome");
                execute(TFB_1());
        }
}
```

Basic mapping:

- text file (cluster) = module
- test case = test case
- keyword = function call
- literal parameter = actual value
- parameter variable = variable with its definition

Mapping on TTCN-3

cluster	TFL_basics		
author	Erik Altena		
version	1.4		
	Application		
start application	Home Bank		
	User	Password	
log in	EA1234	welcome	
test case	TFB_1	HBTC	HBSC
	Component	Runs on	System
map	self	opPortClient	opPortSystem
	Balance		
get balance	&Keep([float] start_bal)		
	To account	Amount	Description
transfer money	123.45.678	25,00	Borrowed
transfer money	234.56.789	653,88	Rent
	Balance		
check balance	&startbal - 678,88		
	Component	Runs on	System
unmap	self	opPortClient	opPortSystem
end test case			
log out			
	Close the application manually after failure!		
	Application		
stop application	Home Bank		

implemented in separate module

support for distr. testing

explicit data type

support for distr. testing

explicit end

Some challenges:

- implicit ends of structures
- no real support for distributed testing
- implicit data types
- pre-defined keywords
- clusters can call each other
- data transfer

Mapping on TTCN-3

cluster	Work
author	Erik Altena
version	1.0
	Cluster
do cluster	Get_coffee
do cluster	Change_spreadsheet
do cluster	Lunch
do cluster	Get_coffee
do cluster	Attend_meetings

```
module Work {
    import from Keywords all;
    import from TestSystemDef all;
    import from SupportTFL all;

    ControlFromWork() {
        tfl_header_cluster("Work");
        tfl_header_author("Erik Altena");
        tfl_header_version("1.0");
        ControlFromGet_coffee();
        ControlFromChangeSpreadsheet();
        ControlFromLunch();
        ControlFromGet_coffee();
        ControlFromAttend_meetings();
    }

    control () {
        ControlFromWork();
    }
}
```

```
module Attend_meetings {
    import from Keywords all;
    import from TestSystemDef all;
    import from SupportTFL all;

    testcase Progress
    runs on PrTC
    system PrSC {
        ...
    }

    ControlFromAttend_meetings() {
        ...
        execute(Progress());
    }

    control () {
        ControlFromAttend_meetings();
    }
}
```

Mapping clusters on modules gave a problem in clusters calling each other.

Module control parts cannot call each other, but functions can.

When each module control only calls a control function, these control functions can call each other.

Mapping on TTCN-3

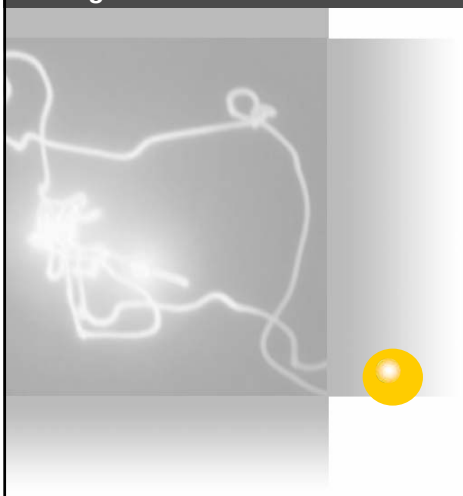
cluster	TFL_basics
author	Erik Altena
version	1.4
start application	Application Home Bank
log in	User EA1234
test case	TFB_1
map	Component self
get balance	Balance &Keep([floa
transfer money	To account 123.45.678
transfer money	234.56.789
check balance	Balance &startbal - 6
unmap	Component self
end test case	
log out	
	Close the a
stop application	Application Home Bank

```
module TFL_basics
{
        import from Actionwords all;
        import from TestSystemDef all;
        import from SupportTFL all;

        testcase TFB_1()
        runs on HBTC
        system HBSC
        {
                var float start_bal;

                map(self:opPortClient, system:opPortSystem);
                get_balance(start_bal);
                transfer_money("123.45.678", "25,00", "Borrowed");
                transfer_money("234.56.789", "653,88", "Rent");
                check_balance(startbal - 678,88);
                unmap(self:opPortClient, system:opPortSystem);
        }

        function ControlFromTFL_basics()
        {
                tfl_header_cluster("TFL_basics");
                tfl_header_author("Erik Altena");
                tfl_header_version("1.4");
                start_application("Home Bank");
                log_in("EA1234", "welcome");
                execute(TFB_1());
                log_out();
                stop_application("Home Bank");
        }

        control
        {
                ControlFromTFL_basics();
        }
}
```

TFL additions:

- explicit ends of structures
- explicit data type declaration
- incorporating test system info (extra test case parameters and new pre-defined keywords)

Agenda



Introduction
TestFrame method and TTCN-3

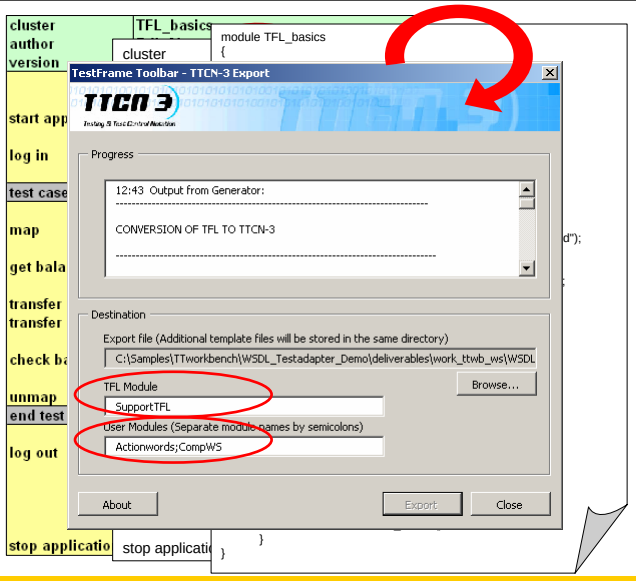
TestFrame Language
keyword-driven

Mapping on TTCN-3
mapping language elements

Translation demo
the bigger picture

Further work
future, conclusions

Translation demonstration



TestFrame clusters can automatically be translated into TTCN-3 modules.

Pre-defined keywords are provided by including a TFL-module.

User-defined keywords have their user-defined function implementation.

Agenda

Introduction
TestFrame method and TTCN-3

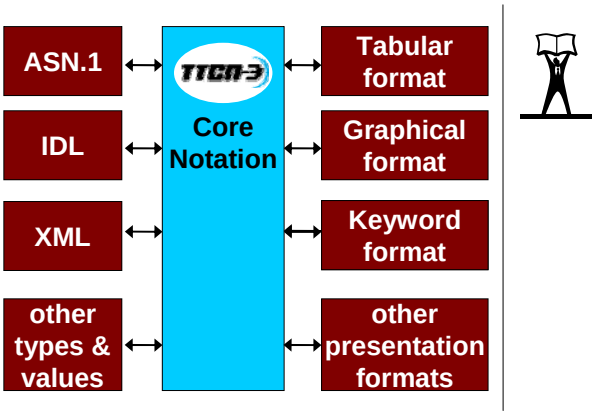
TestFrame Language
keyword-driven

Mapping on TTCN-3
mapping language elements

Translation demo
the bigger picture

Further work
future, conclusions

Further work

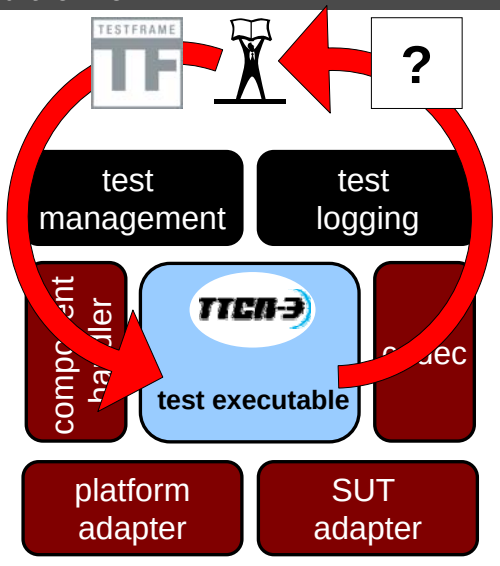


For a complete keyword-driven test presentation format the language mapping must be bi-directional.

All TTCN-3 language elements could be supported in the keyword format.

A subset however will increase readability.

Further work



TestFrame test specification is on a higher abstract level.

After translation to TTCN-3 and execution, the logging/reporting should have the same higher abstract level.

Further work

TestFrame separates test logic from its technical implementation.

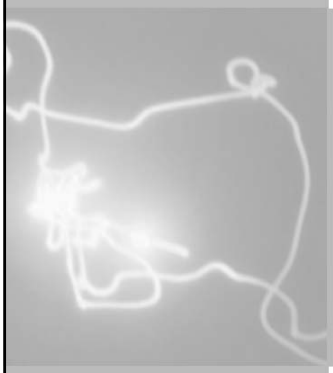
TTCN-3 separates test specification from its implementation.

Combined, TTCN-3 could separate test logic from technical specification and implementation.

Conclusions

- TTCN-3 can have a presentation format on a high-level of abstraction.
- A keyword-driven presentation format would increase re-usability.
- More work needs to be done on the bi-directional mapping.
- More work needs to be done on the logging format.

Thank you!



E-mail

Erik.Altena@LogicaCMG.com

Jos.van.Rooyen@LogicaCMG.com