An Overview of ASD

Formal Methods in daily use

FM2009 Fringe Meeting

Guy H. Broadfoot CTO Verum BV

guy.broadfoot@verum.com

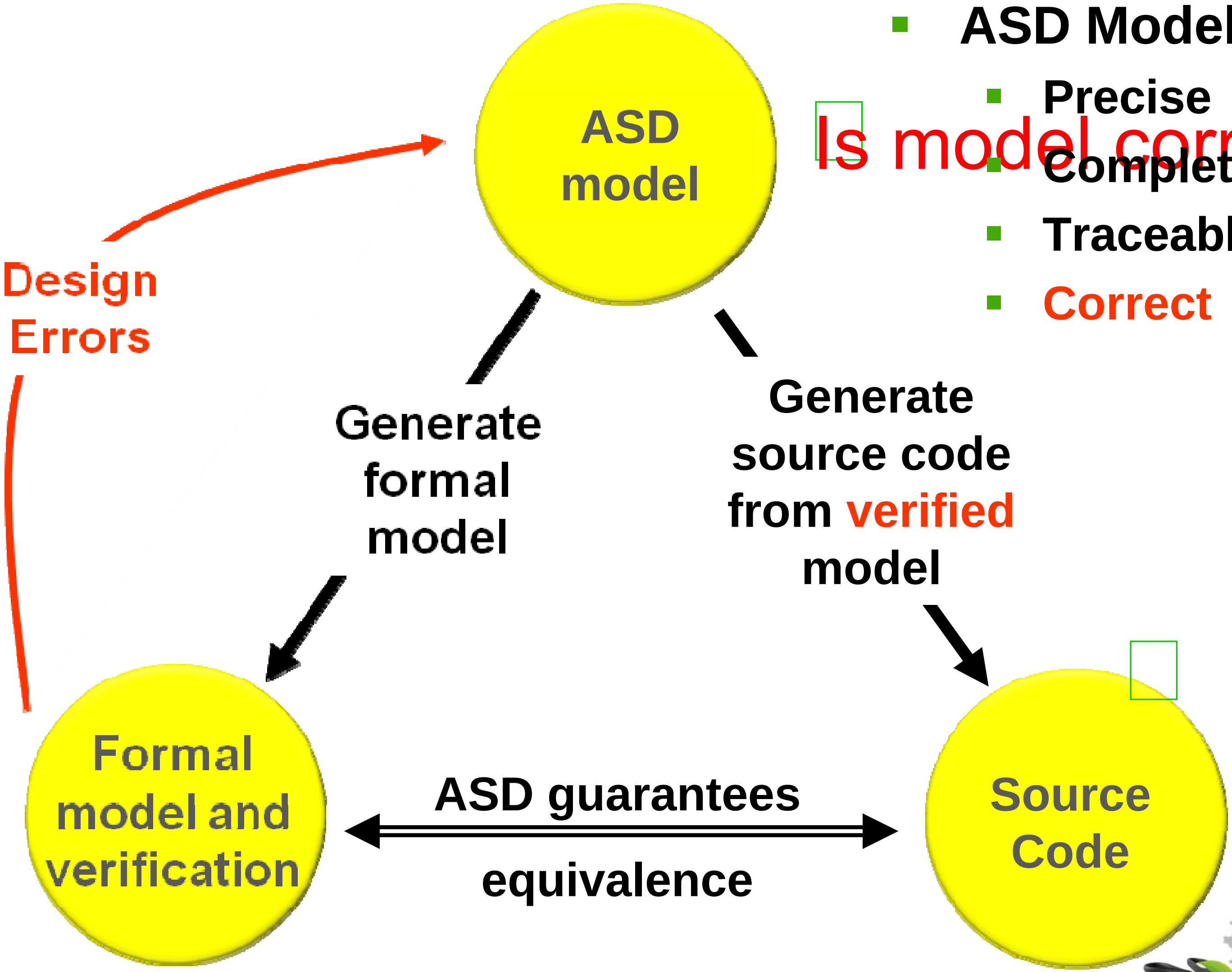
Requirements for gaining industry acceptance

- Must be usable by existing Software Engineers
  - No complex notations
  - No new mathematical skills required
  - Fully automated verification
  - No big changes to existing process or infrastructure
- Must be scalable to industrial sized systems
- Must have a strong business case
  - Shorter time to market
  - Lower costs
  - Reduce delivered defects
  - Reduce the number of people required
- Must have a quick payback
  - Breakeven on first project

What is ASD?

- ASD is a software engineering tool for:
  - constructing complete and correct industrial scale systems from components *formally verified during design*
- ASD provides:
  - fully automated formal verification of specifications and designs
  - fully automatic code generation (C++, C#, MISRA C, Java)
  - easy integration into existing software development teams
- ASD guarantees:
  - behavioural equivalence between specifications, designs, formal models and runtime behaviour of generated code
- ASD is a *paradigm shift*
  - software engineers make specifications and design models and formally verify them instead of coding and testing

ASD is a Paradigm Shift



- ASD Models are:
  - Precise
  - Complete
  - Traceable
  - Correct

Is model correct ?

- MISRA C
- C++
- C#
- Java

Is source code correct ?

ASD Concepts

- A component is a common unit of
  - Functional Specification
  - Design
  - Verification
  - Code generation
  - Runtime execution
- Interface Models
  - Implementation free specification of externally visible behaviour
  - Independent of target programming language
- Design Models
  - Implementation of all internal behaviour and interactions with used components
  - Inherits its own implemented interface model and uses the interface models of the used components
  - Target programming language independent

Don’t cut development… Cut cost

ASD Technologies

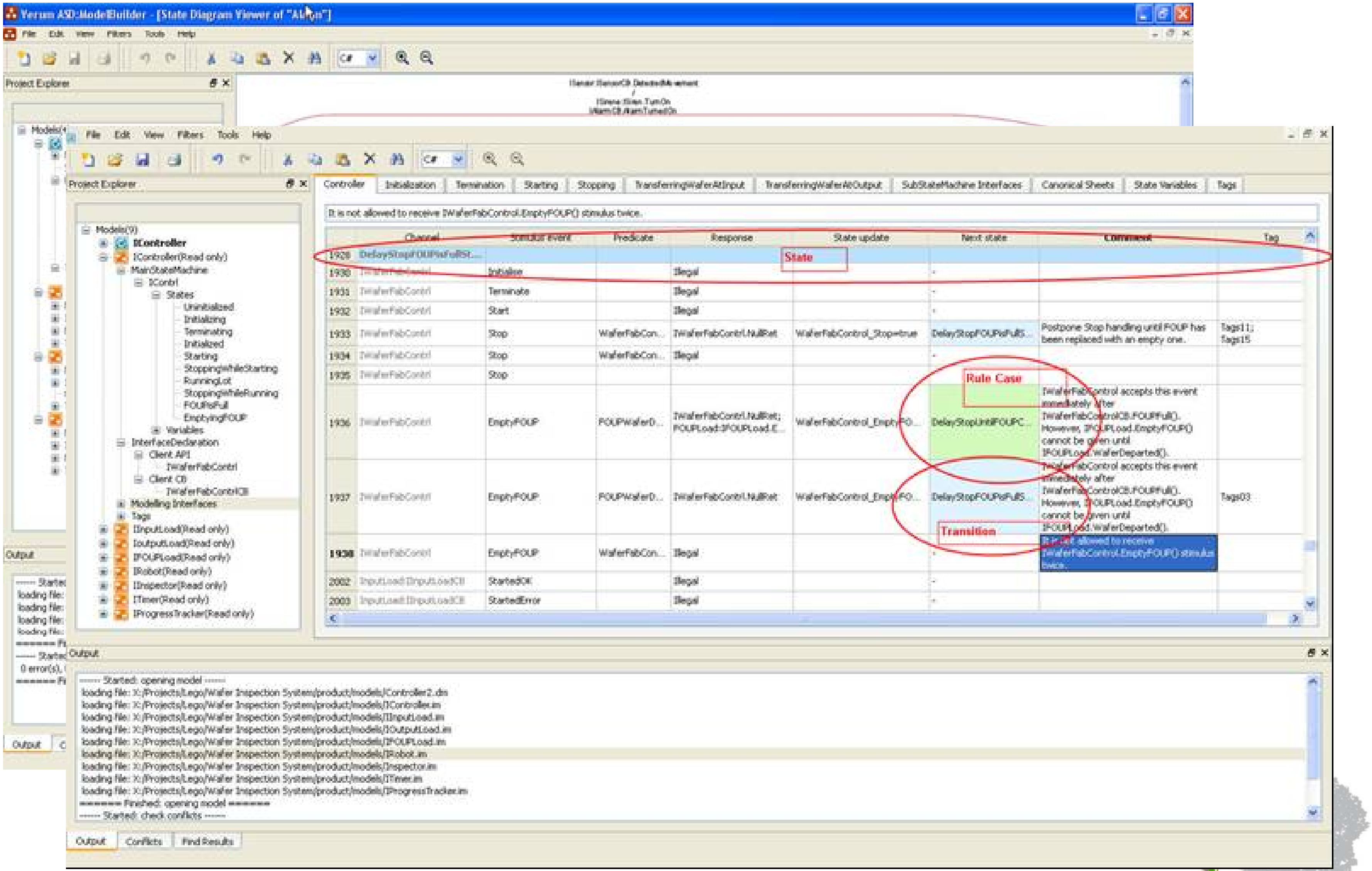
- Sequence-based Specification
  - Basis of ASD Modelling Language
  - Draws on regular expressions and Mealy machines
- CSP + FDR used for verification
- ASD Runtime Model
  - Gives operational semantics to SBS
  - Rules for translating ASD models to CSP
  - Rules for translating ASD models to target programming languages
  - Target language specific ASD Runtime

What is Verified?

- Checks on every Interface Model (implemented and used interfaces)
  - Predicates are well-formed and complete
  - Divergence free
  - Deadlock free
- Checks on the Design Model
  - Design Model must be Deterministic
  - Design must comply with Used Component Interfaces (illegal use of interface, race conditions)
- Design + Used Component Interface Models + Queue
  - Predicates in design model must be complete and well formed
  - All state variables in design model must be in range
  - All use of UCV variables in design model must be valid
  - No Queue overflow
  - Design + Used Components + Queue must be deadlock free
  - Design + Used Components + Queue must be divergence free
  - Design + Used Components + Queue must fully and correctly implement its specification

ASD Model Builder

verum°



It is not allowed to receive IWaferFabControl.EmptyFOUP() stimulus twice.

	Channel	Stimulus event	Predicate	Response	State update	Next state	Comment	Tag
1929	DelayStopFOUPisFullSt...							
1930	IWaferFabContrl	Initialize		Illegal		-		
1931	IWaferFabContrl	Terminate		Illegal		-		
1932	IWaferFabContrl	Start		Illegal		-		
1933	IWaferFabContrl	Stop	WaferFabCon...	IWaferFabContrl.NullRet	WaferFabControl_Stop=true	DelayStopFOUPisFullS...	Postpone Stop handling until FOUP has been replaced with an empty one.	Tags11; Tags15
1934	IWaferFabContrl	Stop	WaferFabCon...	Illegal		-		
1935	IWaferFabContrl	Stop						
1936	IWaferFabContrl	EmptyFOUP	FOUPWaferD...	IWaferFabContrl.NullRet; FOUPLoad:IFOUPLoad.E...	WaferFabControl_EmptyFO...	DelayStopUntilFOUPC...	IWaferFabControl accepts this event immediately after IWaferFabControlCB.FOUPFull(). However, IFOUPLoad.EmptyFOUP() cannot be given until IFOUPLoad.WaferDeparted().	
1937	IWaferFabContrl	EmptyFOUP	FOUPWaferD...	IWaferFabContrl.NullRet	WaferFabControl_EmptyFO...	DelayStopFOUPisFullS...	IWaferFabControl accepts this event immediately after IWaferFabControlCB.FOUPFull(). However, IFOUPLoad.EmptyFOUP() cannot be given until IFOUPLoad.WaferDeparted().	Tags03
1938	IWaferFabContrl	EmptyFOUP	WaferFabCon...	Illegal		-	It is not allowed to receive IWaferFabControl.EmptyFOUP() stimulus twice.	
2002	InputLoad:IInputLoadCB	StartedOK		Illegal		-		
2003	InputLoad:IInputLoadCB	StartedError		Illegal		-		

```
------ Started: opening model ------
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/Controller2.dm
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/IController.im
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/IInputLoad.im
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/IOutputLoad.im
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/IFOUPLoad.im
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/IRobot.im
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/Inspector.im
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/ITimer.im
loading file: X:/Projects/Lego/Wafer Inspection System/product/models/IProgressTracker.im
======= Finished: opening model =======
------ Started: check conflicts ------
```

verum°

ASD:Suite Adopters

PHILIPS
sense and simplicity

Healthcare

Philips Applied Technologies

Nanda Tech

SIOUX EMBEDDED SYSTEMS

ccm
centre for concepts in mechatronics

NXP founded by Philips

logica

HOGESCHOOL ROTTERDAM

overtref jezelf

ICT Automatisering
ADVANCED THINKING

FEI COMPANY™
TOOLS FOR NANOTECH

TASS
software professionals

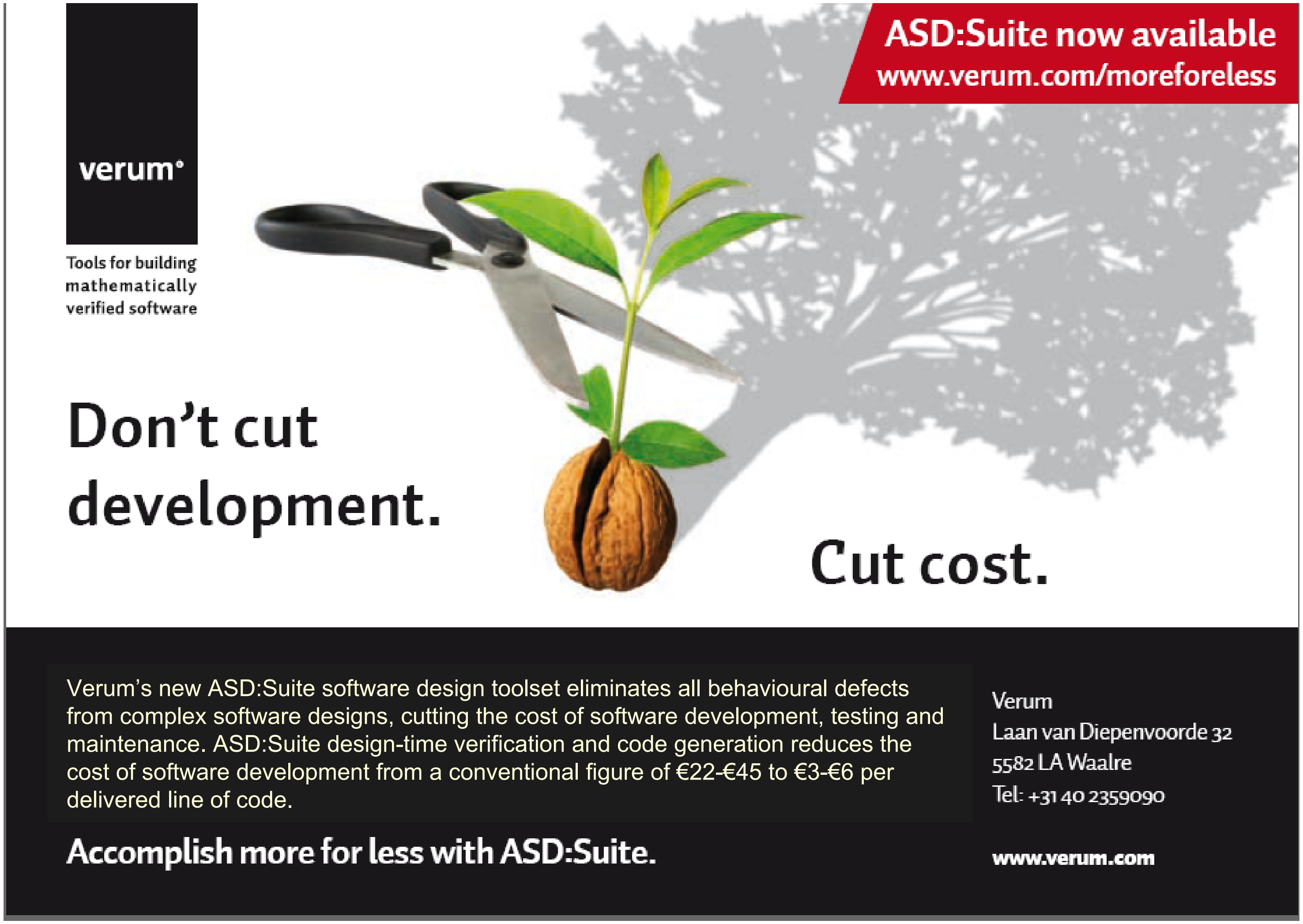
ASM®
The Process of Innovation℠

Fontys
Hogescholen

Nucletron
Improving patient care

Don't cut development… Cut cost

Questions?



verum°

Tools for building mathematically verified software

ASD:Suite now available
www.verum.com/moreforeless

Don’t cut development.

Cut cost.

Verum’s new ASD:Suite software design toolset eliminates all behavioural defects from complex software designs, cutting the cost of software development, testing and maintenance. ASD:Suite design-time verification and code generation reduces the cost of software development from a conventional figure of €22-€45 to €3-€6 per delivered line of code.

Accomplish more for less with ASD:Suite.

Verum
Laan van Diepenvoorde 32
5582 LA Waalre
Tel: +31 40 2359090

www.verum.com

Come and see more in room 118