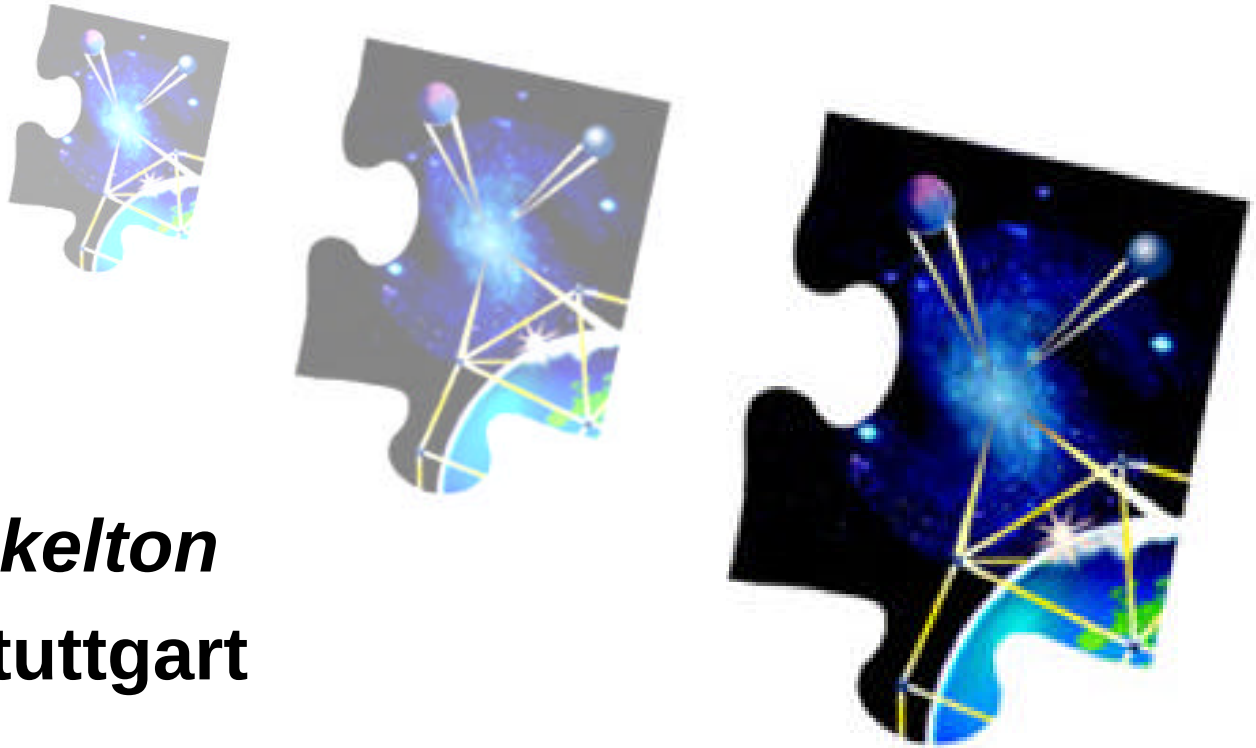
Cutting the Cost of Complexity

- SDL & Requirements of Signalling Systems



William H. Skelton

SOLINET, Stuttgart

SOLINET

Contents

- Background
- Signalling Requirements
- Possible Evolution

SOLINET

SDL History

- Origins as a graphical representation
- Mapping from presentation to execution
- Enhanced to ‚compete' with C++ & OO
- Consolidation
  - Market forces
  - Technology management

SOLINET

SDL Strong Points

- Graphical Representation
  - Intuitive, easy to learn
  - System Organisation & Execution Flow
- Concept of States & Signals
  - FSM specific
  - Simplifies validation & diagnosis

SOLINET

SDL Weak Points

- Complexity affects tools & users
  - Tools are complex
  - Users need skill profiles
- SDL, TTCN & MSC evolved separately
  - Validation & Maintenance out of scope
- C, C++, C#, Java have evolved faster
  - Commercial tools are more powerful

SOLINET

Signalling Systems

- Based on FSMs
  - Independent specialist domain
- High reliability
- Intense competition
- Wide-range of applications
  - Communicating embedded systems
- Relatively long life-cycle

SOLINET

Typical SDL Functionality

- Hierarchical Instantiation & Connectivity
  - Static, not dynamic
- States, Signals, Timers, Variables
- Types (Bit Oriented & ASN.1)
  - Encoding, decoding & formatting

SOLINET

Work Processes

- Implementation
  - Implement, Validate, Maintain
  - Specify, Design, Code, Execute
- Validation
  - Specify, Design, Code, Execute
- Maintenance
  - Diagnose, Analyse, Implement, Validate

SOLINET

Information Content

- Implementation
  - Intended behaviour
- Validation
  - Expected behaviour
  - Actual behaviour under test
- Maintenance
  - Actual behaviour in live use

SOLINET

Information Reuse

- Implementation
  - Component libraries (stacks, types)
- Validation
  - Interfaces (Signals, Parameters)
  - Preambles, postambles, procedures
  - Subset of intended behaviour
- Maintenance
  - Subset of intended behaviour

SOLINET

Meth

- Implementation SDL (& MSC)
- Validation TTCN (& MSC)
- Maintenance MSC
- SDL, TTCN, MSC Overlap
- Information reuse is not implicit
  - Format conversion needed
  - De facto isolation of work processes

SOLINET

Typical TTCN Functionality

- As per SDL plus
- Test Suites
  - Structure & Test Purposes
- Constraints
  - Expected values for received parameters
- Verdicts
  - Pass, Fail, Inconclusive

SOLINET

Typical MSC Functionality

- As per SDL (sub-set) plus
- Signal transmission order
- Parameters
- Timing
- Strong potential for automated checks
  - Link actual and intended behaviour

SOLINET

Possible Evolution

- Markets are driving evolution
  - Survival of most efficient technology
  - Efficient technology needs efficient tools
  - Efficient tools need efficient methodologies
- Convergence of Methodologies
  - Implementation, validation & maintenance
  - MSC & TTCN overlap strongly with SDL
  - UML may be a wrapper for SDL

SOLINET

Outlook

- SDL ‘03 Conference
  - Stuttgart, 1st-4th July 2002
  - Back to Basics
  - Telecoms & Automotive Applications
- SDL Design Contest
  - Simplest, validated design
  - Traffic Light Controller
- www.SDL-FORUM.org

SOLINET